

VYSOKÁ ŠKOLA BÁŇSKÁ – TECHNICKÁ UNIVERZITA OSTRAVA
EKONOMICKÁ FAKULTA

KATEDRA APLIKOVANÉ INFORMATIKY

Racionalizácia databázového systému pre ubytovaciu spoločnosť
Rationalization of Database System for an Accommodation Company

Student: Dominika Gajdičiarová
Vedoucí bakalářské práce: Ing. Vítězslav Novák, Ph.D.

Ostrava 2012

Zadání bakalářské práce

Student: **Dominika Gajdičiarová**
Studijní program: B6209 Systémové inženýrství a informatika
Studijní obor: 6209R001 Aplikovaná informatika
Téma: **Racionalizácia databázového systému pre ubytovaciu spoločnosť**
Rationalization of Database System for an Accommodation Company

Zásady pro vypracování:

1. Úvod
 2. Teoretické východiska tvorby databáz
 3. Analýza současného stavu, návrh řešení a jeho realizace
 4. Zhodnotení navrhovaného řešení
 5. Závěr
- Zoznam použitej literatúry
Zoznam skratiek
Prehlásenie o využití výsledkov bakalárskej práce
Zoznam príloh
Prílohy

Seznam doporučené odborné literatury:

VIESCAS, John a Jeff CONRAD. *Microsoft Office Access 2007 Inside Out*. Redmond: Microsoft Press, 2007. ISBN 0-7356-2325-2.
KRUCZEK, Aleš. *Microsoft Office Access 2010: Podrobná uživatelská příručka*. Brno: Computer Press, 2010. ISBN 978-80-251-3289-0.
MOLINARO, Anthony. *SQL: Kuchařka programátora*. Přeložil Jakub MUŽÍK. Brno: Computer Press, 2009. ISBN 978-80-251-2617-2.


Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.


Vedoucí bakalářské práce: **Ing. Vítězslav Novák, Ph.D.**

Datum zadání: 25.11.2011

Datum odevzdání: 11.05.2012




Ing. Petr Rozehnal, Ph.D.
vedoucí katedry


prof. Dr. Ing. Dana Dluhošová
děkanka fakulty

Prehlasujem, že som celú prácu, vrátane všetkých príloh vypracovala samostatne.

.....


Dominika Gajdičiarová

V Ostrave dňa 11.5.2012

Rada by som na tomto mieste poďakovala vedúcemu bakalárskej práce Ing. Vítězslavu Novákovi, za jeho odborné rady, ďalej firme K-Service, za ochotu pri spolupráci a v neposlednom rade mojej rodine a priateľovi za podporu pri spracovaní tejto práce.

Obsah

1	Úvod.....	5
2	Teoretické východiská tvorby databáz.....	6
2.1	Databáza	6
2.2	Relačný model dát (RMD)	6
2.3	Relačná databáza	7
2.3.1	Primárny kľúč.....	7
2.3.2	Cudzí kľúč	8
2.3.3	Entitno-relačný diagram (ERD):	8
	Prvky entitno-relačného diagramu:	8
2.3.4	Normalizácia	9
2.4	Integritné obmedzenie (IO)	12
2.5	Systém riadenia bázy dát	13
2.5.1	Coddové pravidlá pre SRBD	14
2.6	Jazyk SQL.....	16
2.7	Microsoft Access	17
2.7.1	Komponenty	17
2.7.2	Microsoft Access 2010	19
2.7.3	Visual Basic for Applications	20
3	Analýza súčasného stavu, návrh riešenia a jeho realizácia	22
3.1	Analýza súčasného stavu	22
3.1.1	Povinnosti ubytovateľa.....	22
3.1.2	Aktuálny stav.....	22
3.1.3	Dostupný software.....	23
3.2	Návrh riešenia.....	25
3.2.1	Design.....	25
3.2.2	Funkčnosť.....	26
3.3	Realizácia riešenia	28
3.3.1	Návrh databáze	28
	Lineárny zápis typov vzťahov	29
3.3.2	Aplikácia	30
4	Zhodnotenie navrhovaného riešenia	46
5	Záver	47

Zoznam použitej literatúry

Zoznam skratiek

Zoznam obrázkov

Zoznam tabuliek

Prehlásenie o využitíu výsledkov bakalárskej práce

Zoznam príloh

1 Úvod

Ľudia si už od obdobia z pred priemyselnej revolúcie uchovávali kolekcie dát podobné databázam. Typickou formou zaznamenania údajov bola papierová tabuľka, ktorá reprezentovala nejaký objekt reálneho sveta, napr. osobu. Stĺpce potom reprezentovali vlastnosti objektu, ako napr. meno, vek, bydlisko. Riadky predstavovali jednotlivé objekty. Takýto spôsob zaznamenávania údajov na papier je prehľadný len pre veľmi malé množstvo údajov. S postupným pribúdaním záznamov sa tabuľka stáva neprehľadnou a vyhľadávanie v nej je veľmi zdĺhavé, pretože musíme prehľadávať údaje sekvenčne, keďže nie sú zotriedené. Problém nastáva aj pri zmene údajov, kedy sa musia údaje v stĺpcoch prepísať, alebo úplne odstrániť, resp. vyškrtnúť. Inou, neskôr používanou, formou uchovávania údajov je aj kartotéka. Tu jeden záznam predstavuje jeden kartotečný list. Tieto záznamy je už možné triediť, napr. v našom prípade podľa mena osoby. Pri zotriedení podľa mena, by ale vyhľadanie všetkých osôb pod osemnásť rokov opäť predstavovalo sekvenčné prechádzanie údajov. Okrem tohto problému tu existuje aj mnoho ďalších, ako napr. možnosť nedodržania konzistencie a integrity údajov. Ani kartotéky teda nie sú najideálnejším spôsobom uchovávania dát. Na pomoc nám prichádzajú informačné technológie, konkrétne databázové systémy. Pomocou nich môžeme dáta efektívne uchovávať, triediť a spravovať.

Táto práca sa zaoberá racionalizáciou databázového systému pre ubytovaciu spoločnosť K-Service. Databázový systém v takýchto spoločnostiach slúži na komplexnú správu rôznych údajov, ako je evidencia rezervácií izieb, ubytovaných host'ov a pod. Rozsiahlejšie systémy v sebe zahŕňujú prepojenie s účtovníctvom, reštauračnými službami, barom, či externým skladom. Firma K-Service však využíva zastaraný spôsob uchovávania dát pomocou papierovej evidencie, podrobnejšie viď. sekcia 3.1.2. Mojou úlohou bolo teda analyzovať požiadavky spoločnosti a podľa nich potom navrhnúť kompletne riešenie databázového systému, v prostredí programu MS Access, ktorý by zjednotil a vylepšil ich spôsob vedenia dát.

2 Teoretické východiská tvorby databáz

Hlavným teoretickým východiskom je zoznámenie sa so základnou teóriou databáz, ktorá je veľmi dôležitá pre akúkoľvek ďalšiu prácu v oblasti databázových systémov, či už v role užívateľa, návrhára alebo programátora informačného systému. [6]

2.1 Databáza

Pojem databáza dnes určite nie je nikomu cudzí. Ľudia majú potrebu evidovať a zhromažďovať informácie už odnepamäti. Celá dnešná moderná spoločnosť je postavená na databázových systémoch, od evidencie občanov, cez zdravotníctvo, hospodárstvo, školstvo, až po letectvo, výskum alebo sieť mobilných telefónov. [6]

V najjednoduchšom slova zmysle je databáza kolekcia záznamov a súborov, ktoré sú organizované za určitým účelom. [1]

Vývoj databáz (krátky skok do histórie)

Vráťme sa asi o 50 rokov naspäť. Práve v 60. rokoch minulého storočia vzniká súčasný pojem databáza, entita, atribút entity a väzba medzi entitami (viď. nasledujúce kapitoly.)

V tejto dobe vzniká ešte jeden pojem, a to databázový model. Ten bol zavedený matematikmi, ako prostriedok na popis databázy. Spočiatku sa používali modely dvojakeho typu: hierarchický (založený na modelovaní hierarchie medzi entitami so vzťahmi podriadenosti a nadriadenosti) a sieťový (vychádza z teórie grafov, kde uzly v grafe odpovedajú entitám a orientované hrany definujú vzťahy medzi entitami). V 70. rokoch sa uvedené databázové modely ukázali byť nedostatočné, a preto vznikol relačný model, ktorý sa stal štandardom a používa sa dodnes. [6]

2.2 Relačný model dát (RMD)

RMD je založený na matematickej teórii množín a predikátovej logike, pričom definuje spôsob reprezentácie dát, spôsob ich ochrany (integritné obmedzenie) a možné operácie nad dátami. RMD navrhol a jeho pravidlá publikoval Dr. E. F. Codd (pracovník firmy IBM) r. 1970 v článku: *"A relational model of data for large shared databanks"*.

Okrem základných definícií, vychádzajúcich z teórie množín a predikátovej logiky, boli v článku obsiahnuté tieto idey:

- RMD oddeľuje dáta, ktoré sú chápané ako relácie od ich implementácie.
- Symetrický prístup k dátam. Pri manipulácii s dátami nás nezaujímajú prístupové metódy k dátam v reláciách.
- Pre prácu s dátami máme k dispozícii relačný kalkul a algebru (matematické aparáty, pomocou ktorých je možné popísať sémantiku (význam konštrukcií) relačných jazykov).
- Pre obmedzenie redundancie dát, máme k dispozícii pojmy pre normalizáciu relácií, čo znamená vhodne navrhovať databázové štruktúry.[10]

2.3 Relačná databáza

Relačné databázy sú databázy vychádzajúce z relačného modelu dát, v ktorom sú dáta uložené v tzv. „reláciách“. Od svojich predchodkýň založených na sieťovom a hierarchickom modeli sa líšia tým, že sú budované na pevnom teoretickom základe. Ako už bolo spomenuté, prvým, kto sa definíciami relácií a relačných databáz zaoberal, bol matematik Dr. E. F. Codd.

Pojem relácia teda pochádza z teórie relačných databáz. Reláciu, si môžeme predstaviť ako tabuľku, čo je dvojrozmerná dátová štruktúra, ktorá sa skladá zo stĺpcov a riadkov. Stĺpce tabuľky reprezentujú položky (polia, atribúty) a riadky predstavujú jednotlivé databázové záznamy (vety). Súbor tabuliek (relácií), ktoré sú medzi sebou rôzne previazané, potom tvorí celú databázu (relačná schéma).[11]

Relačné databázy sa vyznačujú predovšetkým tým, že sú relatívne jednoduché, ľahko pochopiteľné a vďaka tomu rozšírené, a medzi programátormi a vývojármi informačných systémov veľmi obľúbené. [6]

2.3.1 Primárny kľúč

Aby mohol byť každý riadok relácie presne identifikovaný, musí byť v relácii stanovená položka, resp. skupina položiek, ktorých hodnoty jednoznačne určujú každý riadok danej tabuľky, tzv. „primárny kľúč“. Z dôvodu efektívnosti sa odporúča, aby primárny kľúč tvoril takú množinu položiek, ktoré spĺňajú podmienku unikátnosti (neexistuje v relácii dve a viac n-tíc, ktoré majú rovnaké hodnoty položiek) a majú minimálnu dĺžku (napr. číselná postupnosť, rodné číslo). [11]

2.3.2 Cudzí kľúč

Súvisiace tabuľky sa navzájom prepájajú pomocou tzv. „cudzích kľúčov“. Cudzí kľúč predstavuje položku, resp. skupinu položiek, ktorých hodnoty sú hodnotami primárneho kľúča inej tabuľky. Prepájanie záznamov pomocou cudzích kľúčov sa považuje za jednu z hlavných predností relačných databáz, pretože odbreňuje používateľa od poznania fyzického uloženia záznamov.[11]

2.3.3 Entitno-relačný diagram (ERD):

ERD zaviedol a prvýkrát použil Peter Pin Shan Chen v roku 1976. Veľmi rýchlo sa rozšíril a stal sa všeobecne uznávaným štandardom. Cieľom ERD je vytvoriť dátový model opisujúci určitú časť reálneho sveta. [16]

Prvky entitno-relačného diagramu sú:

- tabuľka (entita) – objekt, ktorý možno jednoznačne identifikovať,
- položky entity (atribúty) – bližšie určujú entitu,
- domény – pravidlá, ktoré obmedzujú prijateľné hodnoty atribútu,
- relácia – vzťah medzi tabuľkami. [11]

Entity sa v týchto diagramoch popisujú pomocou obdĺžnikov, vzťahy sa znázorňujú pomocou kosoštvorcov a atribúty pomocou elíps.

Typy vzťahov (relácie)

1:1 – vzniká medzi dvomi tabuľkami, kde jedna hodnota primárneho kľúča v hlavnej tabuľke odpovedá hodnote poľa v druhej tabuľke (napr. tabuľka objednávky, v ktorej každý záznam má väzbu na tabuľku s vyúčtovaním: každá objednávka odpovedá jednému vyúčtovaniu).

1:N – vzniká medzi dvomi tabuľkami, kde jedna hodnota primárneho kľúča v hlavnej tabuľke odpovedá niekoľkým hodnotám poľa v druhej tabuľke (napr. vzťah medzi tabuľkou so zoznamom kupujúcich a tabuľkou objednávky: každý kupujúci totiž môže urobiť niekoľko objednávok, ale naopak každá objednávka patrí jedinému kupujúcemu).

M:N – vzniká medzi dvomi tabuľkami, kde každý záznam (je jedno z ktorej z týchto tabuliek) môže odpovedať niekoľkým záznamom z druhej tabuľky (napr. vzťah medzi

objednávkami a produktmi: každá objednávka totiž môže obsahovať niekoľko produktov alebo produkt sa môže nachádzať v niekoľkých objednávkach). [11]

2.3.4 Normalizácia

Normalizácia ER modelu je súbor pravidiel, ako by sme mali postupovať pri transformácii štruktúry entít a relácií ER modelu na štruktúru fyzického usporiadania tabuliek a vzťahov v databázy. [8]

Jedná sa o proces, pri ktorom sa tabuľky rozkladajú za účelom jednoduchšej práce s dátami, ich lepšej manipulácie, zabráneniu redundancie dát (dátá, ktorá sa zbytočne opakujú), lepšej konzistencie dát (viď 1.4) a zabráneniu tzv. aktualizáčnym anomáliám. Tento proces vedie k prehľadnejšej, rozšíriteľnejšej a výkonnejšej databáze. Normalizácia by mala viesť k vzniku tabuliek, ktoré je možné ľahko udržiavať a efektívne sa na ne dotazovať. Normalizovaná schéma musí zachovať všetky závislosti pôvodnej schémy a tabuľky musia zachovať pôvodné dáta, čo znamená, že sa musíme pomocou prirodzeného spojenia dostať k pôvodným dátam.

Normalizácia ale nie je proces, ktorý by zároveň znamenal navýšenie výkonu databázy. Taktiež to nie je úloha, ktorá by bola deterministická, takže existuje viac „správnych“ riešení. Normálne formy sú teda pravidlá, ktoré by dáta v tabuľke mali spĺňať. Čím vyššia normálna forma, tým lepšia a jednoduchšia by mala byť práca s dátami, ich vyberaním a aktualizáciami. Formy idú postupne za sebou od nižších k vyšším, pričom každá vyššia forma v sebe zahŕňa formy nižšie. [7]

Normálne formy:

1NF – Prvá normálna forma

2NF – Druhá normálna forma

3NF – Tretia normálna forma

BCNF – Boyce Coddova normálna forma

4NF – Štvrtá normálna forma

5NF – Piata normálna forma

1. normálna forma (1NF)

Prvá, najjednoduchšia, normálna forma (značíme 1NF) hovorí, že všetky atribúty sú atomické, to znamená ďalej už nedeliteľné (inými slovami, hodnotou nesmie byť tabuľka). A taktiež platí, že:

- v tabuľke nie sú žiadne opakujúce sa záznamy,
- sú definované všetky kľúčové atribúty,
- všetky atribúty závisia na primárnom kľúči. [8]

2. normálna forma (2NF)

Tabuľka sa nachádza v 2NF, ak je v 1NF a každý neľúčový atribút je plne závislý na primárnom kľúči, a to na celom kľúči, nie len na nejakej jeho podmnožine. Aby to bolo možné, musí sa primárny kľúč skladať z viac, ako jedného atribútu. Pokiaľ primárny kľúč zahŕňa iba jedno pole, tabuľka je automaticky v 2NF, ak spĺňa 1NF. [16]

3. normálna forma (3NF)

V tejto forme sa nachádza tabuľka, ak spĺňa predchádzajúce dve formy a žiaden z jej atribútov nie je tranzitívne závislý na primárnom kľúči. Tranzitívna závislosť je závislosť medzi minimálne dvoma atribútmi a kľúčom, kde jeden atribút je funkčne závislý na kľúči a druhý atribút je funkčne závislý na prvom atribúte.

Iné vyjadrenie hovorí, že tabuľka je v 3NF, pokiaľ je v 2NF a všetky neľúčové atribúty sú navzájom nezávislé.

Boyce Coddova normálna forma (BCNF)

Je poslednou prakticky využívanou formou. BCNF normálna forma sa pokladá za variáciu tretej normálnej formy, a dokonca je pôvodnou definíciou 3.NF, tak ako bola publikovaná v 70 rokoch. Pokiaľ je relácia v BCNF, je i v 3NF. Naopak to však neplatí. Relácia sa nachádza v BCNF, ak pre každú netriviálnu závislosť $X \rightarrow Y$ platí, že X je nadmnožinou nejakého kľúča relácie R . Aby bola porušená BCNF, musí byť splnených niekoľko podmienok, a to pomerne špecifických:

- Relácia musí mať viac kandidátnych kľúčov.
- Minimálne 2 kandidátne kľúče musia byť zložené z viac atribútov.
- Niektoré zložené kandidátne kľúče musia mať spoločný atribút.

BCNF v podstate hovorí, že medzi kandidátnymi kľúčmi nesmie byť žiadna funkčná závislosť.

Existujú ešte dve normálne formy, 4NF a 5NF, tie sa však v praxi moc nevyužívajú. Pre úplnosť uvediem iba ich definície. [8]

Štvrtá normálna forma (4NF)

Tabuľka je v štvrtej normálnej forme, ak je v BCNF a popisuje iba príčinnú súvislosť (jeden fakt). Iná definícia hovorí: „Relácia je v štvrtej normálnej forme, pokiaľ je v BCNF, a navyše všetky závislosti sú zároveň funkčnými závislosťami z kandidátnych kľúčov.“ Iný zdroj uvádza: Štvrtá normálna forma sa zaoberá vzťahmi vnútri zloženého primárneho kľúča. Pokiaľ je v tabuľke zložený primárny kľúč, môže sa stať, že niektoré hodnoty tohto kľúča sú na sebe nezávislé, ale tým, že spolu tvoria kľúč, vzniká falošná súvislosť medzi týmito hodnotami a nemôžu existovať nezávisle na sebe, čo nie je v súlade s modelovanou realitou. 4.NF preto vyžaduje, aby kľúč tvorili iba tie hodnoty, ktoré majú skutočnú vzájomnú súvislosť.

Piata normálna forma (5NF)

Relácia je v piatej normálnej forme, pokiaľ je v štvrtej a nie je možné do nej pridať ďalší atribút (skupinu atribútov) tak, aby sa vplyvom skrytých závislostí rozpadla na niekoľko čiastkových relácií. Inak povedané: piata normálna forma sa týka primárnych kľúčov, ktoré sú tvorené najmenej tromi atribútmi. V prípade, že medzi týmito hodnotami v kľúči existujú párové cyklické závislosti, tak je treba tieto závislosti extrahovať do samostatných tabuliek, ale pôvodnú tabuľku je v niektorých prípadoch treba zachovať! [8][16]

2.4 Integritné obmedzenie (IO)

Inak nazývané aj konzistencia. Relačný model dát špecifikuje štruktúru dát v databáze, ale k tomu, aby sme mohli používať databázu ako zdroj dát, je nutné zaistiť, aby sa do nej dostali iba dáta, ktoré tam patria a nestratili sa dáta, ktoré by nemali. Taktiež je potrebné mať k tomu určité mechanizmy. Týmito mechanizmami sú integritné obmedzenia. Databáza, resp. dáta, sú konzistentné, pokiaľ sú v stave vyhovujúcom IO. To znamená, že sa žiadnou manipuláciou dát (úpravou, zmazaním), dáta neztratili alebo nepoškodili. Taktiež v databáze nie sú dáta, ktoré tam nemajú čo robiť, napríklad zoznam kontaktov zmazaného užívateľa a pod. Preto existujú integritné obmedzenia, ktoré majú podobným nehodám zabrániť. Vzhľadom k tomu, že k porušeniu integrity databázy môže dôjsť niekoľkými spôsobmi, rozoznávame niekoľko druhov integritných obmedzení.

Entitné

Ide o špecifikáciu primárneho kľúča tabuľky. Toto obmedzenie implementuje každá relačná databáza a pri pokuse o vloženie dát, nevyhovujúcich tomuto obmedzeniu, dôjde ku chybe.

Doménové

Doménová integrita znamená, že na úrovni stĺpcov definujeme obmedzenie na určitý dátový typ, prípadne obmedzenie rozsahu hodnôt. Inak povedané je to obor hodnôt, ktorých môže atribút nadobúdať. V databázovom žargóne sa tomu hovorí aj doména atribútu.

Referenčné

Referenčná integrita je už medzitabuľková záležitosť. Definuje vzťah dvoch tabuliek, a to pomocou cudzích kľúčov. Tabuľky, ktorých sa tento vzťah týka, bývajú anglicky označované ako master, detail alebo parent, dependent. Cudzí kľúč v tabuľke určuje atribút (skupinu atribútov), ktoré majú buď hodnotu NULL, alebo hodnotu primárneho kľúča niektorého riadku nadradenej tabuľky. [9]

2.5 Systém riadenia bázy dát

Databázový systém (DBS - Database System) je program, ktorý sa používa na vytvorenie, udržiavanie, modifikovanie a správu relačnej databáze. Mnoho ich taktiež obsahuje nástroje na vytváranie aplikácií pre koncového užívateľa. Je to teda softwarové vybavenie, ktoré zaisťuje prácu s databázou, tzn. tvorí rozhranie medzi aplikačnými programami a uloženými dátami.

DBS je tvorený bázou dát, alebo inak, databázou (DB - Database, iba samotné dáta) a systémom riadenia báze dát SRBD (DBMS - DataBase Managment System).

Schopnosti SRBD:

Aby mohol byť nejaký programový systém označený za SRBD, musí byť jednak schopný efektívne pracovať s veľkým množstvom dát, ale taktiež musí byť schopný riadiť (vkladať, modifikovať, mazať) a definovať štruktúru týchto perzistentných dát (čím sa líši od простého súborového systému).

V súčasnosti majú databázové systémy aj mnoho ďalších charakteristických vlastností:

- podporu pre definíciu dátových modelov (príklad typov dátových modelov: relačný, objektový),
- využitie niektorého jazyka vyššej úrovne pre manipuláciu a definíciu dát (napr. SQL),
- autentizáciu užívateľov a autorizáciu operácii nad dátami,
- správu transakcií,
- robustnosť a zotaviteľnosť po chybách bez strát akýchkoľvek dát.

Známy výrobcovia:

- IBM – DB2
- Oracle – Oracle, MySQL
- Microsoft – Microsoft SQL Server, Microsoft Access[16]

2.5.1 Coddové pravidlá pre SRBD

Týchto dvanásť pravidiel vychádza z *Coddovej* teoretickej práce na relačnom modeli a vyjadruje skôr ideálny cieľ, ktorý by mal dosiahnuť relačný DBMS (RDBMS):

1. Pravidlo informácie: Všetky informácie v relačnej databáze sa na logickej úrovni reprezentujú explicitne hodnotami v tabuľkách.
2. Pravidlo zaručeného prístupu: Musí byť zaistené, aby úplne každý údaj v relačnej databáze bol logicky prístupný použitím názvu tabuľky, hodnoty primárneho kľúča a názvu stĺpca.
3. Systematické ošetrovanie prázdnych hodnôt: Prázdne hodnoty (nie nuly, či prázdne reťazce) sú systematicky plne podporované RDBMS pre reprezentáciu chýbajúcich informácií a neplatných informácií nezávisle na dátovom type. (Typicky riešené hodnotou NULL).
4. Popis štruktúry založený na relačnom modeli: Popis databázy sa na logickej úrovni reprezentuje rovnako, ako bežné dáta tzn. v tabuľkách, na ktoré sa môžu oprávnení používatelia dotazovať rovnako, ako na akúkoľvek inú tabuľku.
5. Pravidlo komplexného dátového jazyka: Relačné systémy môžu podporovať viac jazykov a režimov prístupu, ale musí existovať minimálne jeden jazyk, ktorého príkazy sú vyjadriteľné nejakou dobre definovanou syntaxou ako reťazce znakov, ktorá podporuje:
 - definíciu dát,
 - definíciu pohľadu,
 - manipuláciu s dátami,
 - obmedzenie integrity,
 - autorizáciu,
 - vymedzenie transakcie.
6. Aktualizácia pohľadu: Všetky aktualizovateľné pohľady je možno aktualizovať systémovo.

7. Vysokoúrovňová manipulácia s dátami: Spracovanie základnej, či odvodenej tabuľky. Aplikuje sa ako jediný operand, či už na vyhľadávanie, ako aj na vloženie a zmenu dát.
8. Fyzická dátová nezávislosť: Aplikácie a terminály zostávajú logicky nedotknuté zmenami v reprezentácii úložiska alebo prístupových metódach.
9. Logická dátová nezávislosť: Aplikácie a terminály sú logicky nedotknuté, pokiaľ sú v tabuľkách prevádzane zmeny v uchovávaní informácií.
10. Nezávislosť integrity: Integritné obmedzenie musí byť definovateľné v dátovom jazyku v databáze, a nie v aplikácií.
11. Distribučná nezávislosť: Databázový jazyk musí byť schopný manipulovať s dátami umiestnenými na inom počítačovom systéme.
12. Pravidlo nenarušenia: Pokiaľ je v systéme viac jazykov, žiadny z nich nesmie mať možnosť manipulovať s dátami v rozpore s integritnými obmedzeniami.[10]

2.6 Jazyk SQL

Jazyk SQL je štandardným nástrojom, z ktorého väčšina dnes používaných databázových systémov vychádza a tieto systémy v rôznej miere tento štandard dodržiajú.

História

História jazyka SQL spadá do 70. a 80. rokov. Prvý štandard bol prijatý v roku 1986 (označovaný ako SQL86). Časom sa však prejavili niektoré nedostatky. Opravená verzia je z roku 1992 a je označovaná ako SQL92. Ten je v oblasti relačných databáz štandardom dodnes. Skratka SQL značí Structured Query Language. Jazyk v sebe zahŕňa nástroje pre tvorbu databáz (tabuliek) a ďalej nástroje na manipuláciu s dátami (vkladanie dát, aktualizácia, mazanie a vyhľadávanie informácií). [6]

SQL patrí medzi tzv. deklaratívne programovací jazyky, čo v praxi znamená, že kód jazyka SQL nepíšeme v žiadnom samostatnom programe, ale vkladáme ho do iného programovacieho jazyka, ktorý je už procedurálny. So samotným jazykom SQL môžeme pracovať len v prípade, že sa terminálom pripojíme na SQL server a v príkazovom riadku zadávame priamo príkazy jazyka SQL.

SQL sa skladá z niekoľkých častí. Niektoré časti sú určené pre administrátorov a návrhárov databázových systémov, iné zase pre koncových používateľov a programátorov. Prvou časťou jazyka SQL je jazyk DDL - Data Definition Language. Jedná sa o jazyk na vytváranie databázových schém a katalógov. Spôsob ukladania tabuliek definuje jazyk SDL - Storage Definition Language. Treťou časťou pre návrhára a správcu je jazyk VDL - View Definition Language, určujúci vytváranie pohľadov (pohľad si môžeme predstaviť ako virtuálnu tabuľku zloženú z rôznych iných tabuliek). Poslednou časťou, je jazyk DML - Data Manipulation Language, ktorý obsahuje základné príkazy INSERT, UPDATE, DELETE a najpoužívanejší príkaz SELECT. S jazykom DML pracujú najviac koncoví užívatelia a programátori databázových aplikácií. [6]

2.7 Microsoft Access

MS Access je relačný databázový systém. Je súčasťou balíka MS Office, avšak nie je súčasťou všetkých jeho verzií. MS Access má podobný vzhľad ako iné produkty balíka Office, zahŕňajúc rozloženie aplikácie a navigáciu, kde však podobnosť končí. Access mal do verzie 2007 koncovku súborov *.mdb*, od tejto verzie sa začína používať koncovka *.accdb*. Ďalším faktom je, že, skoršie verzie nedokážu spracovávať súbory s touto koncovkou, ale naopak MS Access 2007 a vyššie dokážu spracovávať staršie súbory. Databázu MS Access *.accdb* môžeme používať buď ako samostatnú RDMBS na jednej pracovnej stanici alebo v zdieľanom *klient-server* móde cez sieť. [1] [12]

2.7.1 Komponenty

Databáza MS Access sa skladá zo siedmich hlavných komponentov: tabuliek, formulárov, zostáv, dotazov, reportov, makier, modulov.

Tabuľky

Databázová tabuľka vyzerá podobne ako hárok, v ktorom sa údaje ukladajú do riadkov a stĺpcov. Dôsledkom toho možno hárok obyčajne jednoducho importovať do tabuľky databázy. Hlavným rozdielom medzi uchovávaním údajov v hárku a v databáze je spôsob organizácie údajov. Každý riadok v tabuľke sa nazýva záznam. V záznamoch sú uložené jednotlivé informácie. Každý záznam pozostáva z jedného alebo viacerých polí. Polia korešpondujú so stĺpcami v tabuľke. Polia musia byť určené ako určitý typ údajov, ako je text, dátum alebo čas, číslo alebo iný typ.

Formuláre

Predstavujú rozhranie, ktoré sa používa pri práci s údajmi a často obsahujú príkazové tlačidlá na vykonávanie rôznych príkazov. Databázu je možné vytvoriť bez formulárov jednoduchou úpravou údajov v údajových hárkoch tabuliek. Väčšina používateľov databáz však na zobrazenie, zadávanie a úpravu údajov v tabuľkách uprednostňuje používanie formulárov. Formuláre poskytujú jednoduchý formát na prácu s údajmi, pričom je k nim možné pridať funkčné prvky, ako sú napr. príkazové tlačidlá. Tlačidlá možno naprogramovať na určovanie, ktoré údaje sa zobrazia vo formulári, na otváranie iných formulárov alebo zostáv, alebo na vykonanie množstva iných úloh.

Zostavy

Zostavy sa používajú na zhrnutie a prezentovanie údajov z tabuliek. Každá zostava sa môže naformátovať, aby prezentovala informácie v čo najčitateľnejšej podobe. Zostavu možno spustiť kedykoľvek, pričom bude vždy odrážať aktuálne údaje v databáze. Zostavy sa všeobecne formátujú tak, aby ich bolo možné vytlačiť. Môžu sa však tiež zobrazíť na obrazovke, exportovať do iného programu alebo odoslať ako e-mailová správa.

Dotazy

Dotazy sú výkonné súčasti databázy a môžu vykonávať množstvo rôznych funkcií. Ich najbežnejšou funkciou je získať špecifické údaje z tabuliek. Údaje, ktoré chceme zobrazíť, sú zvyčajne roztrúsené v rôznych tabuľkách a dotazy umožnia ich zobrazenie v jednom údajovom hárku. Keďže nie vždy chceme vidieť naraz všetky záznamy, dotazy umožňujú pridať kritériá na filtrovanie len požadovaných záznamov. Dotazy často slúžia ako zdroj záznamov pre formuláre a zostavy.

Makrá

Makrá v programe Access možno považovať za zjednodušený programovací jazyk, ktorý sa môže použiť na pridanie funkcií do databázy. Môžete napríklad pripojiť makro k príkazovému tlačidlu na formulári, aby sa makro spustilo pri kliknutí na tlačidlo. Makrá obsahujú akcie, ktoré vykonávajú úlohy, ako otvorenie zostavy, spustenie dotazu alebo zatvorenie databázy. Väčšina manuálnych databázových operácií sa môže pomocou makier zautomatizovať, čím sa výrazne ušetrí čas. Makro ale nedokáže vyhodnocovať rôzne vstupy, teda rozhodovať sa.

Moduly

Moduly sú, podobne ako makrá, objekty, ktoré sa používajú na pridanie funkcií do databázy. Zatiaľ čo makrá sa v programe Access vytvárajú zo zoznamu akcií makier, moduly sa píšú v programovacom jazyku Visual Basic for Applications (VBA), vid' 1.7.2. Modul je súhrn deklarácií, príkazov a procedúr, ktoré sú uložené spolu ako jednotka. Modul môže byť buď modul triedy, alebo štandardný modul. Moduly tried sú priložené k formulárom alebo zostavám a zvyčajne obsahujú procedúry špecifické pre formulár alebo zostavu, ku ktorej sú priložené. Štandardné moduly obsahujú všeobecné procedúry, ktoré nie sú priradené k žiadnemu inému objektu. [4]

2.7.2 Microsoft Access 2010

Program Microsoft Access 2010 je hlavne o jednoduchosti používania. Obsahuje šablóny pripravené na použitie a výkonné nástroje, ktoré sa postarajú o relevantné výsledky aj pri postupne narastajúcom množstve údajov. Nové webové databázy, ktoré boli pridané do programu Access, ďalej rozširujú možnosti práce s údajmi tým, že zjednodušujú sledovanie a používanie údajov a ich zdieľanie s ostatnými používateľmi. [2]

Hlavné výhody :

Rýchlejšie a jednoduchšie vytváranie databáz

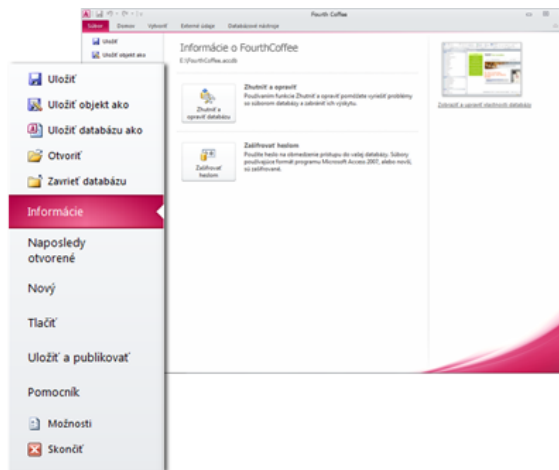
Šablóny pripravené na okamžité použitie a opakovane použiteľné súčasti, ktoré sú v programe Access 2010, robia z tohto programu rýchle a jednoduché databázové riešenie.

Vytváranie pôsobivejších formulárov a zostáv

V podmienenom formátovaní je teraz možné používať údajové pruhy a pravidlá podmieneného formátovania možno spravovať v samostatnom intuitívnom zobrazení.

Jednoduchší prístup k nástrojom

Nové zobrazenie Microsoft Office Backstage™ poskytuje priamy prístup k databázovým nástrojom a umožňuje efektívnejšie spravovať databázu. Zobrazenie Backstage nahrádza tradičnú ponuku *Súbor* vo všetkých aplikáciách balíka Office 2010 a predstavuje centralizované usporiadané miesto na vykonávanie úloh súvisiacich so správou databázy a na prispôsobenie možností používania programu Access.



Obrázok 1 – Ukážka interface MS Access 2010

Nové spôsoby prístupu do databáz

Pridaná nová služba Access Services v rámci servera Microsoft SharePoint Server 2010 slúži na sprístupňovanie databáz na webe pomocou novej funkcie webových databáz. [5]

2.7.3 Visual Basic for Applications

Programovací jazyk Microsoft Visual Basic for Applications (skrátene VBA) je efektívny a pre bežné aplikácie plnohodnotný programovací jazyk, implementovaný v prostrediach kľúčových programov kancelárskeho balíčka MS Office, ako sú Excel, Word, Powerpoint, Access, Outlook, a iné, ako napr. jazyk pre programovanie makier.[13]

Väčšina rutinných operácií s tabuľkami, e-mailami, prezentáciami a textom sa dá zautomatizovať použitím jazyka VBA, a to dvoma spôsobmi:

- *Naprogramovaním obslužného makra*, vid' 1.7.1.
- *Naprogramovaním VBA aplikácie* - VBA aplikácie na rozdiel od makier dokážu vyhodnocovať vstupujúce premenné, a na základe vyhodnotenia sa rozhodnúť, ako s nimi ďalej pracovať. VBA aplikácie sú plnohodnotné programy, ktoré dokážu spolupracovať s databázovými servermi, inými aplikáciami, súbormi, vykonávať komplexné výpočty a simulácie.

Aplikácie VBA, a tiež makrá, dokážu vo viacerých oblastiach pomerne efektívne nahradiť robustné programy vytvorené profesionálnymi programátorskými firmami, oproti ktorým majú dokonca viacero výhod:

- jazyk Visual Basic for Applications je implementovaný priamo v programoch Word, Excel, Access, Outlook a Powerpoint, čo značne znižuje náklady na ich vývoj. VBA aplikácie vedia okamžite pracovať napr. s bunkami, funkciami, kontingenčnými tabuľkami v Exceli, či s modulom pre kontrolu pravopisu vo Worde, vďaka čomu sú ich vývoj, rozširovanie, či aktualizácie niekoľkonásobne lacnejšie, ako je vývoj bežných aplikácií.
- Aplikácie VBA nie sú pred uvedením do produkcie preložené do strojového kódu. Program priamo číta programátorom zadaný zdrojový kód. Užívatelia aplikácie si môžu priamo v programe overiť fungovanie jednotlivých funkcií.
- Aplikácie VBA nie je nutné inštalovať. Aplikácie sa môžu distribuovať jednoduchým kopírovaním súboru.

Používanie VBA aplikácií a makier však má, oproti bežným programom, aj svoje nevýhody.

- Hlavnou nevýhodou týchto aplikácií v porovnaní so štandardnými je ich relatívne nižšia rýchlosť. Pri bežných kancelárskych aplikáciách je tento rozdiel zanedbateľný, rozdiel je však významný v prípade vykonávania robustných simulácií, prípadne zložitých numerických výpočtov, kedy by bola nižšia rýchlosť značne na obtiaž.
- VBA aplikácie pracujú priamo so zdrojovým kódom, čo má aj určité nevýhody. Prečítaním zdrojového kódu sa užívateľ dozvie, ako aplikácia pracuje, čo nie je vhodné pri aplikáciách, ku ktorým je nekontrolovaný prístup. Tento fakt predurčuje jazyk VBA pre tvorbu najmä vnútrofirmerých aplikácií.[13]

3 Analýza súčasného stavu, návrh riešenia a jeho realizácia

3.1 Analýza súčasného stavu

V tejto sekcii uvediem povinnosti ubytovateľa, z ktorých budem vychádzať pri tvorbe návrhu riešenia, ďalej aktuálny stav ubytovateľa a dostupný software.

3.1.1 Povinnosti ubytovateľa

Podľa ustanovenia § 3 ods. 4 zákona je ubytovateľ povinný viesť evidenčnú (domovú) knihu. Domová kniha je dokument, do ktorej ubytovateľ zapisuje:

- meno a priezvisko ubytovaného,
- deň, mesiac a rok narodenia,
- štátne občianstvo,
- číslo dokladu,
- začiatok a koniec ubytovania.

Zápisy do domovej knihy musia byť vedené prehľadne a zrozumiteľne. Tieto zápisy musia byť usporiadané postupne, z časového hľadiska. Domovú knihu pre účely kontroly vedie ubytovateľ v písomnej podobe, digitalizovaná podoba je pre účely kontroly vylúčená.

Domovú knihu ubytovateľ uchováva po dobu šiestich rokov od vykonania posledného zápisu.[3]

Viesť domovú knihu je jediná zákonom daná povinnosť, ktorú musí plniť každý vlastník ubytovacieho zariadenia. Ostatné veci týkajúce sa prevádzky, ktoré si firma potrebuje uchovávať, sú individuálne podľa toho, aké služby zariadenie poskytuje. Niektoré poskytujú len základne služby spojené s ubytovaním, iné sú rozsiahlejšie, kedy pod ubytovacie zariadenie spadá napríklad reštaurácia, alebo rôzne športové, či wellness služby a iné.

3.1.2 Aktuálny stav

Firma K-Service je ubytovacia spoločnosť, ktorá vlastní ubytovacie zariadenie v Prahe a hotel vo Veľkých Popoviciach. Poskytuje možnosť krátkodobého ubytovania, ktoré využívajú najmä turisti, a taktiež ubytovania na dlhodobo, ktoré je určené hlavne pre firmy a brigádnikov.

Spoločnosť si vedie záznamy o rezerváciách hostí, knihu ubytovaných, a z nej vyplývajúce poplatky, ktoré musia odvádzať. Taktiež si uchovávajú jednotlivé faktúry a pokladničné doklady, ktoré potom spracúva externá účtovnícka firma. Poplatky, faktúry a pokladničné doklady sú vedené v zjednodušenej elektronickej podobe. Poplatky sú spracovávané v programe MS Excel a faktúry spolu s pokladničnými dokladmi v programe Fakturace 3, ostatné záznamy sú uchovávané v papierovej podobe, čo je veľmi pracné a časovo neefektívne.

Vyhľadanie akejkoľvek informácie prebieha manuálne, čo je neporovnateľne pomalšie oproti programovému vyhľadávaniu údajov v elektronickej databáze. Pracovníci sú síce na tento spôsob práce zvyknutí, ale, už zo spomínaných časových dôvodov, by uvítali automatizované riešenie uchovávania, vytvárania a vyhľadávania údajov. Softwarové riešenie šité na mieru by nielenže ušetrilo čas, ale taktiež by sprehľadnilo vedenie údajov, zefektívnilo celkovú prevádzku a aspoň z časti eliminovalo chyby z nepozornosti, ako napr. rezervácia izby, ktorá je už na dané obdobie obsadená.

Softwarové portfólio v tejto oblasti je naozaj pestré. Na ukážku v stručnosti popíšem niektorých jeho zástupcov.

3.1.3 Dostupný software

Uvedené popisy sú prebraté z [14][15]

Agnis

Program plne využíva možnosti, ktoré ponúka operačný systém Windows a ďalej poskytuje grafické rozhranie, ovládanie najmodernejších periférií a zariadení, väzbu na internet, moderný SQL jazyk a podobne. Základnou črtou programu je názorná grafická informatívna obrazovka o aktuálnej obsadenosti hotelu – ubytovanie, rezervácie a uvedenie mimo prevádzku na jednotlivých izbách (zruboch, chatkách). Z tejto základnej obrazovky je možné intuitívne vstupovať do formulárov poskytujúcich podrobné informácie, vykonávať rôzne spracovania, zmeny stavu, vyúčtovania a vyhodnotenia. Systém obsahuje množstvo modulov, ktoré slúžia na správu účtovníctva, skladových zásob, reštaurácie a pod.

Horec

Horec je moderný, spoľahlivý, bezpečný a používateľsky príjemný systém pre hotelové recepcie, vhodný pre všetky typy ubytovacích zariadení. Splňuje náročné požiadavky chodu recepcie a poskytuje jej pracovníkom vysoký štandard, vrátane možnosti doplnenia o radu ďalších funkcionalít, ako sú odpočet telefónu, rezervácie priestorov, Pay-TV, kľúčové kartové systémy, web-rezervácie, prepojenie na reštauračný a ekonomický systém a pod. Jeho multijazyčnosť a kompatibilita s inými systémami zvyšuje efektivitu práce v každom ubytovacom zariadení. Systém Horec sa vyznačuje vysokou flexibilitou tvorby cenníku na princípe cenových profilov. Grafické prehľady zobrazujú jednotlivé izby, poschodia aj priestory, čím uľahčujú orientáciu recepčných a poskytujú okamžitý prehľad o obsadenosti, host'och, objednávkach a celkovom stave hotelu. Systém Horec je inšpirovaný modernými hotelovými trendmi a vybavený vysoko spoľahlivými a pokrokovými technológiami.

Existuje mnoho ďalších hotelových systémov, ako napríklad:

- ABX Recepce
- Fidelio
- Mefisto
- Hores Plus
- Open Hotel

Spoločným znakom týchto systémov je až prílišná komplexnosť. Spoločnosť K-Service nepotrebuje takéto komplexné riešenie, práve naopak, požaduje, čo najprehľadnejšie a najjednoduchšie, no zároveň kvalitné riešenie, ktoré poskytne len požadované moduly a funkcie.

Všetky tieto požiadavky sa budem snažiť uplatniť v návrhu riešenia, viď. ďalej.

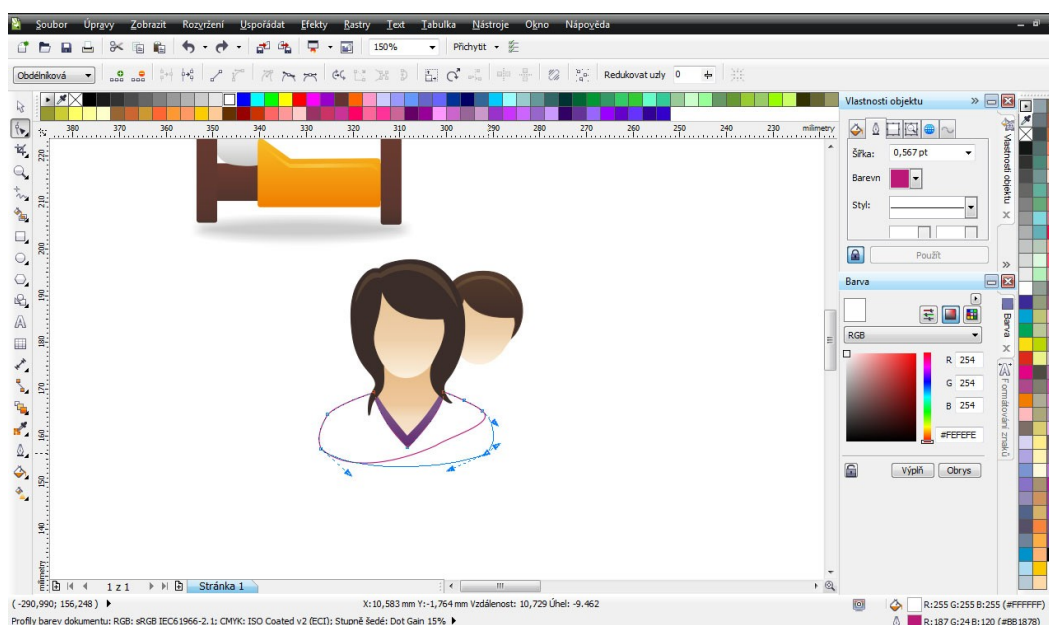
3.2 Návrh riešenia

3.2.1 Design

„Design is the fundamental soul of a human-made creation that ends up expressing itself in successive outer layers of the product or service.“ Steve Jobs

Celý systém je koncipovaný tak, aby bol intuitívny, príjemný a prehľadný. Väčšina ekonomických, či hotelových systémov má často monotónne a veľmi komplexné GUI, čo je možno z časti zapríčinené množstvom modulov a funkcií, ktoré poskytujú, avšak podľa môjho názoru treba aj v takom prípade dbať na celkový „look and feel“ aplikácie. Mojim zámerom, čo sa týka designu, bolo vytvorenie „hravého“, nie však prehnane farebného, či preplneného GUI. Napriek obmedzeným možnostiam Accessu, boli jednotlivé formuláre vytvorené tak, aby obsahovali v čo najväčšej možnej miere všetko potrebné, no zároveň aby boli jednoduché, a aby sa v nich zamestnanec vedel dobre orientovať.

Jednotlivé formuláre sú doplnené rôznymi ikonami, ktoré majú okrem estetickej funkcie aj informačný charakter. Zdrojom a zároveň inšpiráciou pri ich tvorbe mi boli tzv. „icon banks“, kde sú k dispozícii rôzne voľne šíriteľné ikony. Nakoľko mi často nevyhovovala ich veľkosť, musela som ich prekresliť ručne, aby pri zväčšení nestratili svoju kvalitu. K tomuto účelu a rovnako k tvorbe nových ikon som využila vektorový grafický editor CorelDraw.



Obrázok 2 – Ukážka tvorby ikon v programe CorelDraw

Kombináciou pôvodných vznikli rôzne alternatívy finálnych ikon vid'. Obr. 3



Obrázok 3 – Ukážka ikon

Ďalšou časťou návrhu bolo farebne odlišiť jednotlivé logické časti aplikácie pre lepšiu orientáciu používateľa. Ten tak bude vďaka tomu okamžite vedieť v akom oddieli sa momentálne nachádza.

Kombinácie farieb :

- *fialovo-lososová* – pre prácu so zákazníkmi a vecami s tým spojenými, ako napríklad rezervácie, či kniha ubytovaných .
- *modro-biela* – veci spojené so základnou administratívou, ktorá je v systéme implementovaná. Pod túto skupinu spadajú faktúry, príjmové a výdajové pokladničné doklady a poplatky.
- *hnedo-zelená* – pre prácu s izbami



Obrázok 4 – Kombinácie farieb

3.2.2 Funkčnosť

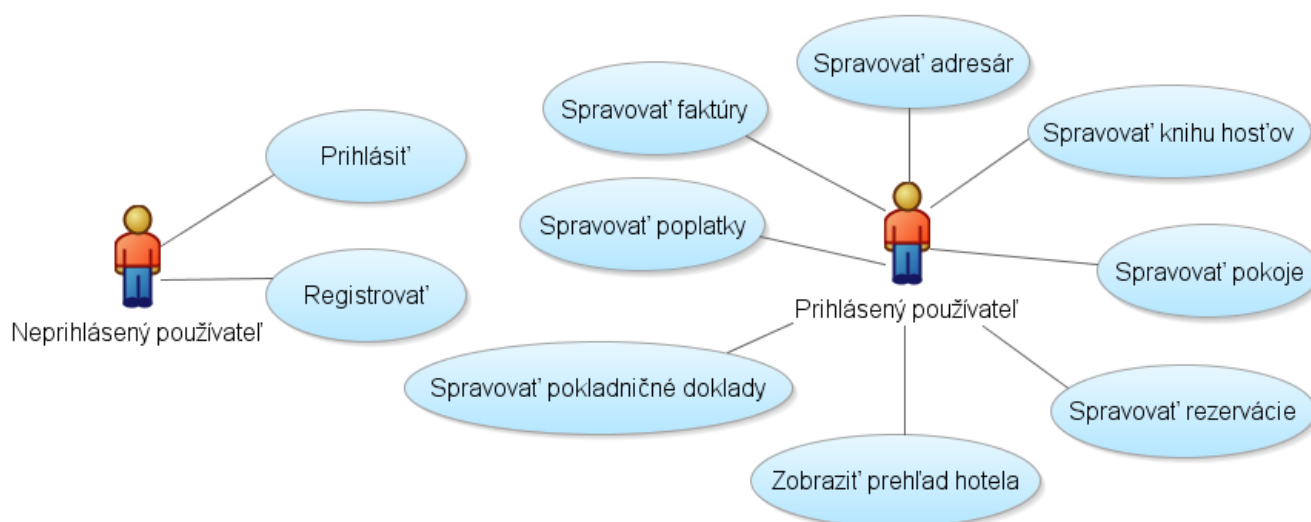
Systém by mal umožňovať prácu len prihláseným používateľom, a to preto, aby bol zabránený prístup nepovolaným osobám a ďalej, aby pri vykonaných zmenách, ako napr. ubytovanie nového hosťa, bolo možné zistiť, kto danú akciu vykonal. Pri danom zázname sa teda uchová aj ID zamestnanca, prípadne jeho meno. Systém samozrejme umožní aj registráciu nového používateľa, avšak tú môže vykonať len oprávnená osoba.

Ďalej by mal byť implementovaný modul slúžiaci na správu ubytovaných hostí. Tu treba dbať na to, že cudzinecká polícia vyžaduje aby boli záznamy o cudzincoch vedené samostatne, alebo nejakým spôsobom odlišené od ostatných. Keďže by mala byť kniha ubytovaných dostupná aj v papierovej podobe, musí byť pridaná možnosť záznamy z tohto modulu vytlačiť.

Požadovaná je tiež implementácia modulu, ktorý sa bude zaoberať správou rezervácií. Tu by malo byť samozrejme umožnené vytvoriť novú rezerváciu, na základe rozpätia dátumov a počtu izieb a v prípade potreby ju zmazať, či editovať. Okrem toho by mal byť dostupný prehľad všetkých rezervácií.

Ako už bolo spomenuté, poplatky má firma spracované v programe MS Excel a faktúry spolu s pokladničnými dokladmi spracúva v programe Fakturace 3. Ponúka sa teda možnosť, buď vytvoriť novú implementáciu pre ich správu, alebo využiť stávajúce riešenie a nejakým spôsobom ho sprístupniť zo systému.

Všetky tieto požiadavky môžeme zhrnúť do nasledovného grafického zobrazenia, vyjadreného pomocou use case diagramu.

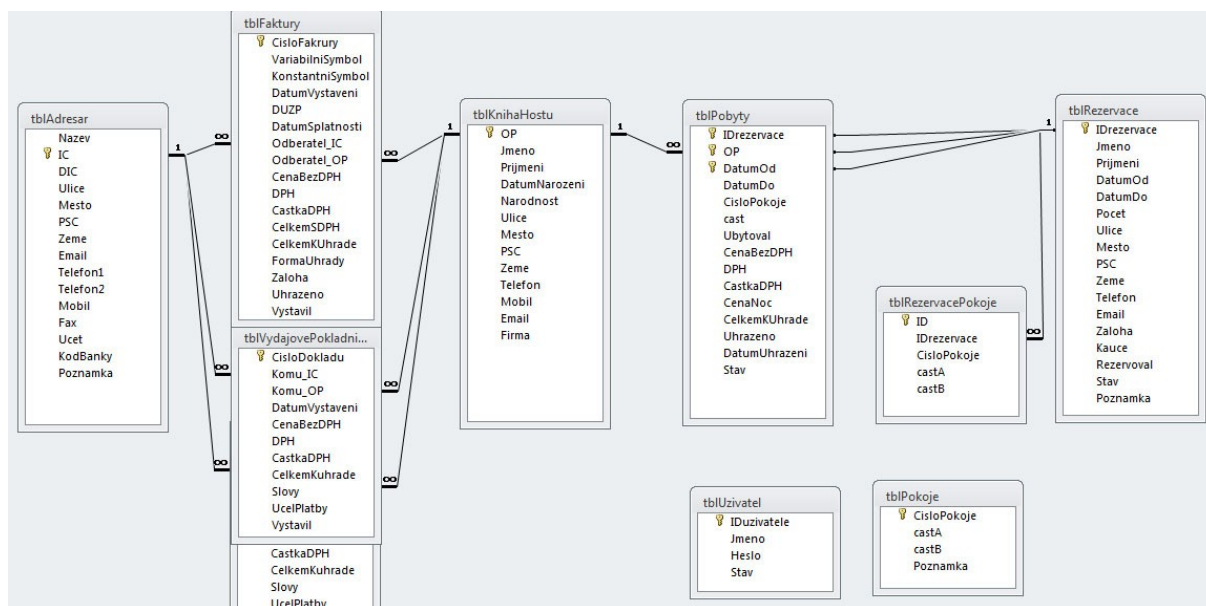


Obrázok 5 – Use case diagram (úroveň 1)

Samozrejme okrem týchto najzákladnejších požiadaviek bude u každého modulu podľa potreby implementovaná i ďalšia funkcionálnosť.

3.3 Realizácia riešenia

3.3.1 Návrh databázy



Obrázok 6 – Diagram vzťahov entít

Lineárny zápis entít (primárny kľúč, cudzí kľúč)

tblPokoje (**CisloPokoje**, castA, castB, Poznamka)

tblUzivatel (**IDuzivatele**, Jmeno, Heslo, Stav)

tblAdresar (Nazev, **IC**, DIC, Ulice, Mesto, PSC, Zeme, Email, Ttelefon1, Telefon2, Mobil, Fax, Ucet, KodBanky, Ponamka)

tblRezervace (**IDrezervace**, Jmeno, Prijmeni, DatumOd, DatumDo, Pocet, Ulice, Mesto PSC, Zeme, Telefon, Email, Zaloha, Kauce, Rezervoval, Stav)

tblPokojeRezervace (**ID**, IDrezervace, CisloPokoje, castA, castB)

tblKnihaHostu (**OP**, Jmeno, Prijmeni, DatumNarozeni, Narodnost, Ulice, Mesto, PSC, Zeme, Telefon, Mobil, Email, Firma)

tblPobyty (**IDrezervace**, **OP**, **DatumOd**, DatumDo, CsiloPokoje, cast, Ubytoval, CenaBezDPH, DPH, CastkaDPH, CenaNoc, CelkemKUhrade, Uhrazeno, DatumUhrazeni, Stav)

tblFaktury (**CisloFaktury**, VyriabilniSymbol, KonstantniSymbol, DatumVystaveni, DUZP, DatumSplatnosti, Odberatel_IC, Odberatel_OP, CenaBezDPH, DPH, CastkaDPH, CelkemSDPH, CelkemKUhrade, FormaUhrady, Zaloha, Uhrazeno, Vystavil)

tblPrijmovePokladniDoklady (CisloDokladu, PrijatoOd_IC, PrijatoOd_OP, DatumVystaveni, CenaBezDPH, DPH, CastkaDPH, CelkemKUhrade, Slovy, UcelPlatby, Vystavil)

tblVydajovePokladniDoklady (CisloDokladu, Komu_IC, Komu_OP, DatumVystaveni, CenaBezDPH, DPH, CastkaDPH, CelkemKUhrade, Slovy, UcelPlatby, Vystavil)

Lineárny zápis typov vzťahov

IZBY_K_REZERVÁCIAM (*tblPokoj*, *tblRezervace*) N:1

REZERVÁCIE-POBYTY (*tblRezervace*, *tblPobyty*) 1:1

HOSTĚ -FAKTURA (*tblKnihaHostu*, *tblFaktury*) 1:N

HOSTĚ -PPD (*tblKnihaHostu*, *tblPrijmovePokladniDoklady*) 1:N

HOSTĚ-VPD (*tblKnihaHostu*, *tblVydajovePokladniDoklady*) 1:N

IČ-FAKTÚRA (*tblAdresar*, *tblFaktury*) 1:N

IČ-PPD (*tblAdresar*, *tblPrijmovePokladniDoklady*) 1:N

IČ-VPD (*tblAdresar*, *tblVydajovePokladniDoklady*) 1:N

POBYTY_HOSTĚA (*tblAdresar*, *tblVydajovePokladniDoklady*) 1:N

Dátový slovník

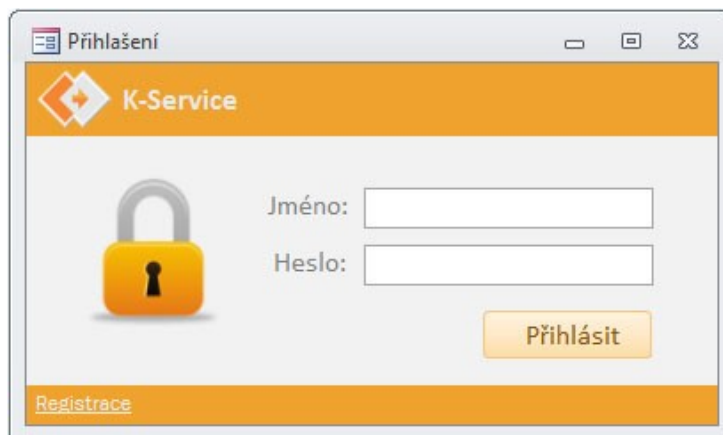
Na ukážku uvádzam dátový slovník tabuľky *tblKnihaHostu*.

Názov	Typ	Veľkosť	Kľúč	Null	Index	Popis
OP	Text	20	PK	Nie	Ano	Občiansky preukaz
Jmeno	Text	30	Nie	Nie	Ano	–
Prijmeni	Text	30	Nie	Nie	Ano	–
DatumNarozeni	Datum a čas	–	Nie	Nie	Nie	–
Narodnost	Text	20	Nie	Nie	Ano	cz/cudzinec
Mesto	Text	50	Nie	Nie	Nie	–
Ulice	Text	50	Nie	Nie	Nie	–
PSC	Text	5	Nie	Nie	Nie	–
Zeme	Text	50	Nie	Nie	Nie	–
Telefom	Text	30	Nie	Ano	Nie	–
Mobil	Text	30	Nie	Ano	Nie	–
Email	Text	30	Nie	Ano	Nie	–
Firma	Text	8	Nie	Ano	Ano	IČ firmy, patrí pod firmu

Tabuľka 1 – Dátový slovník tabuľky *tblKnihaHostu*

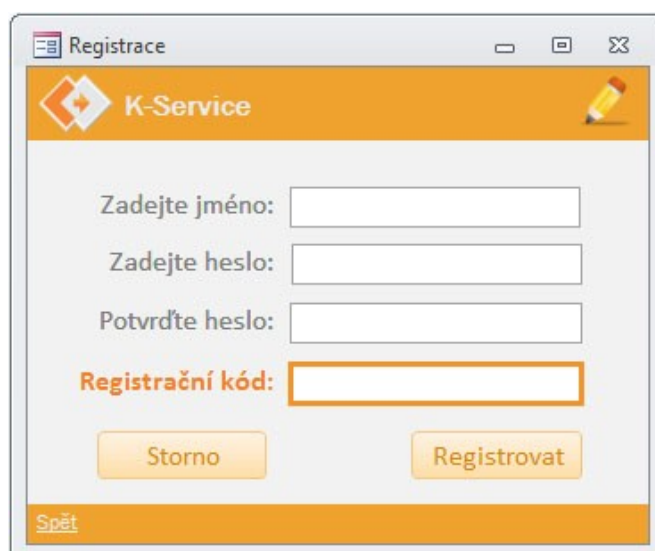
3.3.2 Aplikácia

Po spustení aplikácie sa zobrazí prihlasovacie okno, vid'. Obr.7, kde sa zadávajú prihlasovacie údaje. V prípade korektného vyplnenia je povolený vstup do systému, v opačnom prípade sa na obrazovke zobrazí chybové hlásenie o nesprávnosti zadaných údajov.



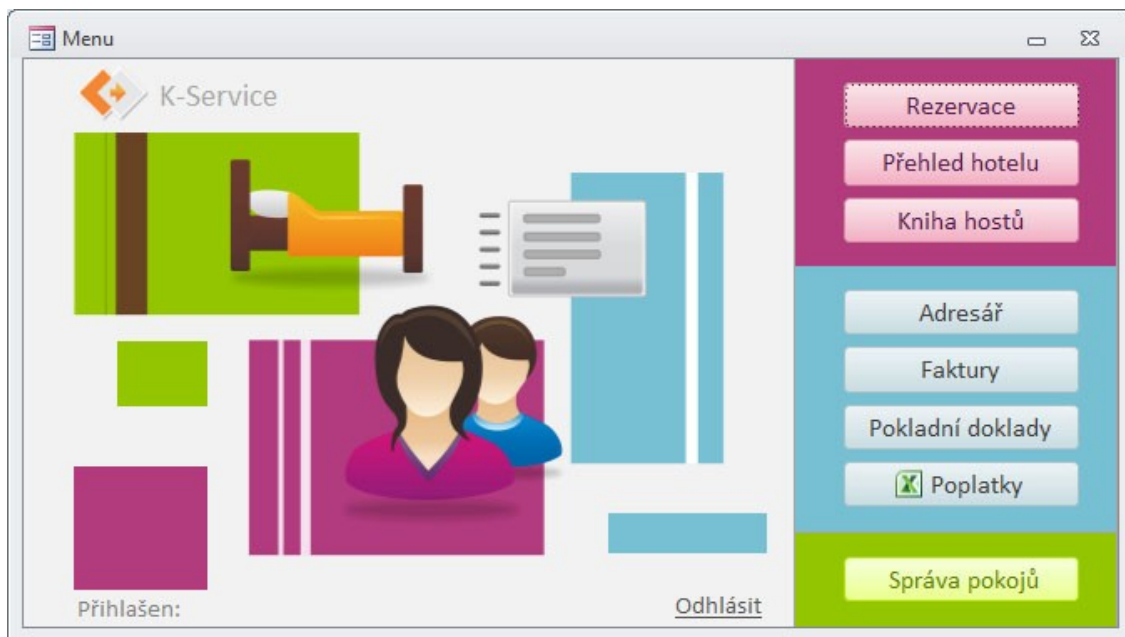
Obrázok 7 – Prihlasovací formulár

Ak užívateľ nie je registrovaný, môže tak učiniť prostredníctvom vyplnenia registračného formulára s tým, že musí zadať *Registračný kód* (bezpečnostný kód), ktorý pozná iba majiteľ ubytovacieho zariadenia. Tu sa dá v rámci možností zabezpečiť, aby so systémom nepracovali nepovolané osoby.



Obrázok 8 – Registrácia používateľa

Po úspešnom prihlásení sa používateľovi zobrazí menu aplikácie, z ktorého môže pristupovať k jednotlivým modulom. Je tu tiež zobrazená informácia o tom, kto je aktuálne prihlásený.



Obrázok 9 – Menu aplikácie

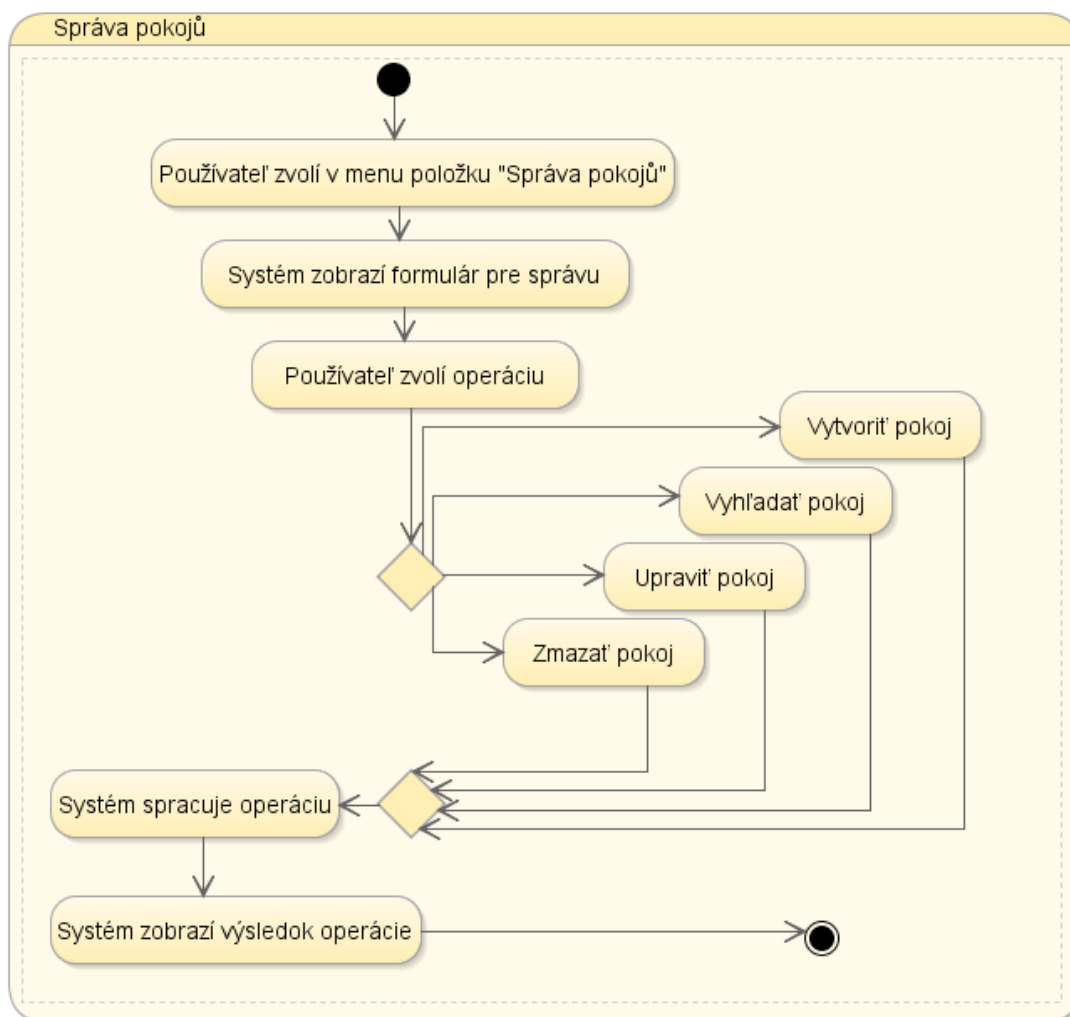
Moduly aplikácie:

- *Správa pokojů*
- *Rezervace*
- *Přehled hotelu*
- *Kniha hostů* → *Pobyty*
- *Adresář*
- *Faktúry*
- *Pokladní doklady*

Ďalej uvediem popis jednotlivých modulov.

Správa pokojů

Modul slouží na správu izieb v ubytovacom zariadení, Tieto izby je možné následne zadávať u rezervácií a ubytovaniach hostí. V tomto module je zobrazený celkový prehľad izieb, ktorého súčasťou je vyhľadávanie. Ďalej je možné v module izby vytvárať, editovať, mazať. Funkčnosť popisuje diagram aktivít na Obr.10



Obrázok 10 – Diagram aktivít: Správa izieb

Po kliknutí na záznam v kompletnom prehľade sa v pravo zobrazia detaily tohto záznamu.

```
Private Sub CisloPokoje_Click()
    Form_frm8PrehledPokoju.RecordsetClone.FindFirst ("[CisloPokoje]='" &
    Form_frm8PrehledPokoju.CisloPokoje & "'")
    Form_frm8Pokoje.Bookmark = Form_frm8PrehledPokoju.RecordsetClone.Bookmark
End Sub
```



Obrázok 11 – Modul: *Pokoje*

Pri vytváraní novej izby sa zadáva číslo izby, poznámka a kapacity oboch jej častí, keďže ubytovacie zariadenie má jednotlivé izby rozdelené na dve časti. Samozrejme sa overuje, či boli zadane všetky potrebné položky a ich korektnosť. Rovnako aj pre ostatné formuláre systému, či už pre vytváranie alebo editáciu. Ďalej pri vytváraní novej izby tiež overujeme, či izba so zadaným číslom už existuje, pomocou:

```

If DCount("[CisloPokoje]", "tblPokoje", "[CisloPokoje] = '" & CisloPokoje & "'") >
0 Then
    MsgBox "Pokoj s číslom: '" & CisloPokoje & "' již existuje!", vbExclamation
    CisloPokoje.SetFocus
    Exit Sub
End If

```

Obrázok 13 – Nová izba

Obrázok 12 – Úprava izby

Pri úprave izby je potrebné pri otvorení formulára uložiť aktuálne údaje, a potom tieto údaje porovnávať s hodnotami získanými po editácií. Na základe zhody, či nezhody sa vykoná príslušná akcia.

Pri odstránení izby treba dbať na to, aby táto izba nefigurovala v rezervácii so stavom: *evidovaná*, alebo *probíhá* (viď. modul **Rezervace** – Stavy rezervácie), a taktiež aby nebola priradená k žiadnemu aktuálne ubytovanému hosťovi. Aby nenastal prípad, kedy hosťovi, či rezervácii ostane pridelená izba, ktorá už neexistuje. Izbu chceme odstrániť zo zdroja dostupných izieb pre rezervácie a ubytovania, napríklad pri rôznych rekonštrukciách. Keďže *cisloPokoje* nie je cudzí kľúč v tabuľkách *tblPobyty* a *tblRezervacePokoje*, tak nie je potrebné záznamy v týchto tabuľkách pred odstránením určitej izby zmazať. Ak by *cisloPokoje* bolo uložené ako cudzí kľúč, museli by byť záznamy so zhodným *cisloPokoje* z týchto tabuliek zmazané a spoločnosť by tak prišla o históriu záznamov. Samozrejme, že keď *cisloPokoje* nie je cudzím kľúčom v týchto tabuľkách, tak treba pred jeho vkladáním do nich overovať, či toto *cisloPokoje* vôbec v *tblPokoje* existuje.

Prehľad izieb môže slúžiť zároveň aj ako zjednodušená forma inventáru, kde sú do poznámky izby vpísané veci, ktoré sa v nej nachádzajú. Ako už bolo spomenuté vyššie, izby definované v tomto module slúžia aj ako zdroj pre rôzne iné akcie v systéme, ako je napríklad vytváranie rezervácie.

Rezervace

Tento modul, viď. Obr.14, poskytuje podrobný prehľad všetkých rezervácií. Používateľ má možnosť jednotlivé rezervácie vyhľadávať, vytvárať, mazať, editovať, či ukončovať. S týmto modulom je úzko spätý modul **Přehled hotelu** (popis tohto modulu je uvedený ďalej), pomocou ktorého môžeme vyhľadať vhodné izby pre danú rezerváciu na základe jej kritérií (napríklad rozsah dátumov a počet osôb). Vygenerované vhodné izby využívame potom ako zdroj pri priradovaní izieb k danej rezervácii viď. ďalej.

Rezervácie sú odlišené na základe statusu (*stavu*). *Stavy* rezervácie:

- *evidovaná* – pri vzniku novej rezervácie
- *probíhá* – po ubytovaní aspoň jedného host'a, ktorý spadá pod rezerváciu
- *ukončená* – pri ukončení rezervácie alebo všetkých jej pobyto(vid'. **Knih**a hostů).

Korektný *Stav* rezervácií sa kontroluje po každom otvorení, prípadne aktualizácii modulu.

Obrázok 14 – Modul: *Rezervace*

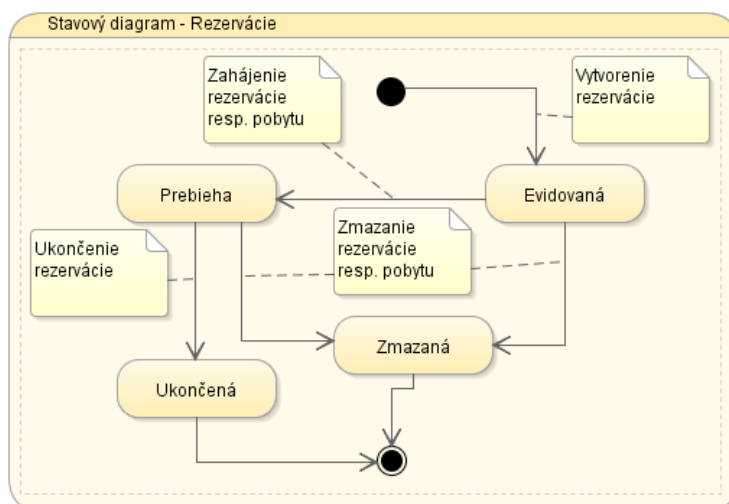
Pri vytvorení rezervácie nemôže ubytovacia spoločnosť žiadať od zákazníka osobné údaje, ako je číslo občianskeho preukazu. Z tohto dôvodu v tabuľke rezervácií rozlíšime jednotlivé osoby na základe *IDrezervace*, prípadne telefónneho čísla, mena, priezviska, emailu, či dátumu rezervácie. Pri vytváraní rezervácie k nej taktiež musíme priradiť izby (spolu s počtom lôžok oboch jej častí), ktoré budú môcť byť využité na samotné ubytovanie host'a resp. hostí. Ako už bolo spomenuté, na generovanie vhodných izieb pre rezerváciu slúži modul ***Přehled hotelu***, ktorého popis je uvedený ďalej. Ak by sa pri vytváraní, alebo editácii pokúsil pracovník priradiť k rezervácii nevyhovujúcu izbu a jej časť (taká, ktorá nespadá do kritérií rezervácie), tak bude na tento fakt upozornený a pridanie nebude umožnené. Za účelom pridania izieb k rezervácii slúži tlačidlo *Pokoje rezervace*, ktoré odkáže používateľa na formulár, kde bude môcť zaznamenať izby a ich časti, ktoré budú priradené k tejto rezervácii. Izba môže byť od rezervácie odstránená len v tom prípade, ak v nej nie je momentálne ubytovaný žiaden hosť.

Rezervácia môže byť zmazaná, pokiaľ je v stave *evidovaná*. Ak je v stave *probíhá*, nemôžeme ju zmazať, môžeme ju len ukončiť.

S ukončením rezervácie sa ukončia aj všetky jej aktívne pobyty (momentálne ubytovaní hostia). Pokiaľ je ukončený nejaký pobyt rezervácie, ktorá okrem neho nemá žiaden ďalší aktívny pobyt, bude používateľovi automaticky ponúknutá možnosť ukončiť celú túto rezerváciu.



Obrázok 15 –Nová rezervácia, úprava rezervácie a pokoje rezervácie



Obrázok 16 – Stavový diagram: Rezervácie

Přehled hotelu

Ako už bolo spomenuté, pomocou tohto modulu môžeme vyhľadávať izby, ktoré vyhovujú zadaným kritériám. Pod tieto kritériá spadá :

- Rozsah dátumov (dátum od – dátum do)
- Počet osôb
- Počet dní

Pokiaľ používateľ zadá len rozsah dátumov, vygeneruje sa prehľad dní a voľných izieb na každý deň v tomto rozpätí dátumov.

Pri každom dátume, resp. dni, v tomto prehľade je zobrazený celkový počet voľných miest v ubytovacom zariadení na tento deň. Po kliknutí na záznam (konkrétny deň) sa zobrazí detailný prehľad jednotlivých voľných izieb. Voľnou izbou sa myslí taká izba, ktorá má aspoň jednu časť neobývanú. Vo vygenerovanom zozname je potom počet u neobývanej časti izby napríklad 3, čo značí jej voľnú kapacitu lôžok a u druhej časti sa zobrazí 0. Z toho vyplýva, že obývaná časť má voľnú kapacitu lôžok 0, bez ohľadu na to koľko ľudí je v nej ubytovaných (1 z 5, 2 z 5, ...).

Ak používateľ pridá k zadanému rozpätiu dní aj počet osôb, zobrazia sa len také dni z tohto rozsahu, kde je počet voľných miest hotela minimálne taký, ako používateľ požaduje. Poslednou možnosťou je kombinácia všetkých troch kritérií. V tomto prípade sa zobrazia len také dni zo zadaného rozsahu, ktoré majú minimálny počet voľných miest taký, ako zadal používateľ a nasledujú v rade za sebou aspoň toľko dní, koľko bolo zadané.

Z formulára prehľadu hotela je pre urýchlenie práce umožnené pristúpiť priamo k formuláru na vytvorenie novej rezervácie.

Datum	Volných míst	Číslo pokoje	Část A	Část B
-------	--------------	--------------	--------	--------

Obrázok 17 – Modul: *Přehled hotelu*

Kniha hostů

Obsahuje prehľad ubytovaných hostí, teda aktuálne ubytovaných, spolu s ich históriou, vid'. Obr.18. Je tu umožnené hostí pridávať, editovať, mazať, filtrovať, či vyhľadávať. V rámci tohto modulu je využitý i ďalší modul **Pobyty**, ktorý slúži na správu pobytov jednotlivých hostí.

Ako bolo spomenuté v návrhu riešenia, je potrebné rozlíšiť ubytovaných hostí na základe národnosti. To dosiahneme pomocou jednoduchého drop-down menu v ich prehľade. Na základe vybranej hodnoty sa vyselektujú daní hostia. Ďalej ich je možné filtrovať aj podľa toho, či majú uhradené svoje pobyty alebo nie. Keďže spoločnosť poskytuje často ubytovanie pre celé firmy dajú sa jednotliví hostia filtrovať aj na základe tohto kritéria, teda firmy, ku ktorej patria. Po kliknutí na záznam v prehľade sú samozrejme zobrazené detaily host'a, ako aj pri ostatných prehľadoch s detailmi v systéme.

The screenshot shows a web application titled 'Kniha hostů'. It features a header with the title and a search filter set to 'všichni'. Below the header is a table listing guests with columns for OP (Occupation), Jméno (Name), Příjmení (Surname), and Národnost (Nationality). To the right of the table, the details for a selected guest (OP: 09093555, Jméno: Dominika, Příjmení: Gajdičiarová) are displayed, including their birth date, nationality, address, and contact information. At the bottom, there are navigation buttons like 'Menu', 'Tisk knihy', 'Smazat', and 'Upravit'.

OP:	Jméno:	Příjmení:	Národnost
09093555	Dominika	Gajdičiarová	cizinec
12313444	Martina	Dorociaková	cizinec
24444422	Gabriela	Lopušánová	cz
24U5634	Ľubomír	Sokolovský	cizinec
35342366	Michal	Puchoň	cizinec
3535354	Lukáš	Obrusník	cz
37534244	Katarína	Tomanová	cz
72552544	Tomáš	Tuhovčák	cizinec
76524424	Martin	Gajdičiar	cizinec

OP: 09093555
Jméno: Dominika
Příjmení: Gajdičiarová
Dat. nar: 1. 1. 1989
Národnost: cizinec
Adresa:
Ulice: Mládeže
Město: Turzovka **PSČ: 02354**
Země: Slovensko
Kontakt:
Telefón:
Mobil: 777666555
E-mail: dg@gmail.com

Obrázok 18 – Modul: *Kniha hostů*

Pri vytváraní nového hosťa (Obr.19) a pri editácii hosťa môžeme údaje vypísať ručne alebo ich importovať z jeho rezervácie. Import je užitočný len vtedy ak zaznamenávame pobyt hosťa, ktorý bol zároveň rezervujúci. To, či už bola daná osoba v zariadení niekedy ubytovaná zistíme z celkového prehľadu modulu **Kniha hostů** na základe čísla občianskeho preukazu, ktorý predkladá hosť pred ubytovaním na recepcii. Dáta, ktoré sa zadávajú pri vytváraní hosťa logicky vyplývajú z tabuľky *tblKnihaHostu*, rovnako aj pri ostatných vytváraníach záznamov u iných modulov.

Odstrániť je možné len takého hosťa, ktorý nemá evidovaný žiaden pobyt, faktúru, či pokladaný doklad. V opačnom prípade, nie je už možné tohto hosťa odstrániť zo systému, keďže ubytovacie zariadenie si musí viesť záznamy o všetkých hosťoch, ktorý v ňom boli ubytovaní.

The image shows two software windows. The larger window on the left is titled 'Upravit' and contains a form for adding a new guest, labeled 'NOVÝ HOST'. The form includes fields for 'Import' (checkbox), 'ID rezervace:' (dropdown), 'OP:', 'Jméno:', 'Příjmení:', 'Národnost:', 'Adresa:' (with sub-fields for 'Ulice:', 'Město:', 'PSČ:', 'Země:'), and 'Kontakt:' (with sub-fields for 'Telefón:', 'Mobil:', 'E-mail:'). At the bottom are 'Storno' and 'Uložit' buttons. A pink arrow points from this window to a smaller window on the right titled 'Úprava hosta:', which has a pink header with 'ÚPRAVA' and a user icon.

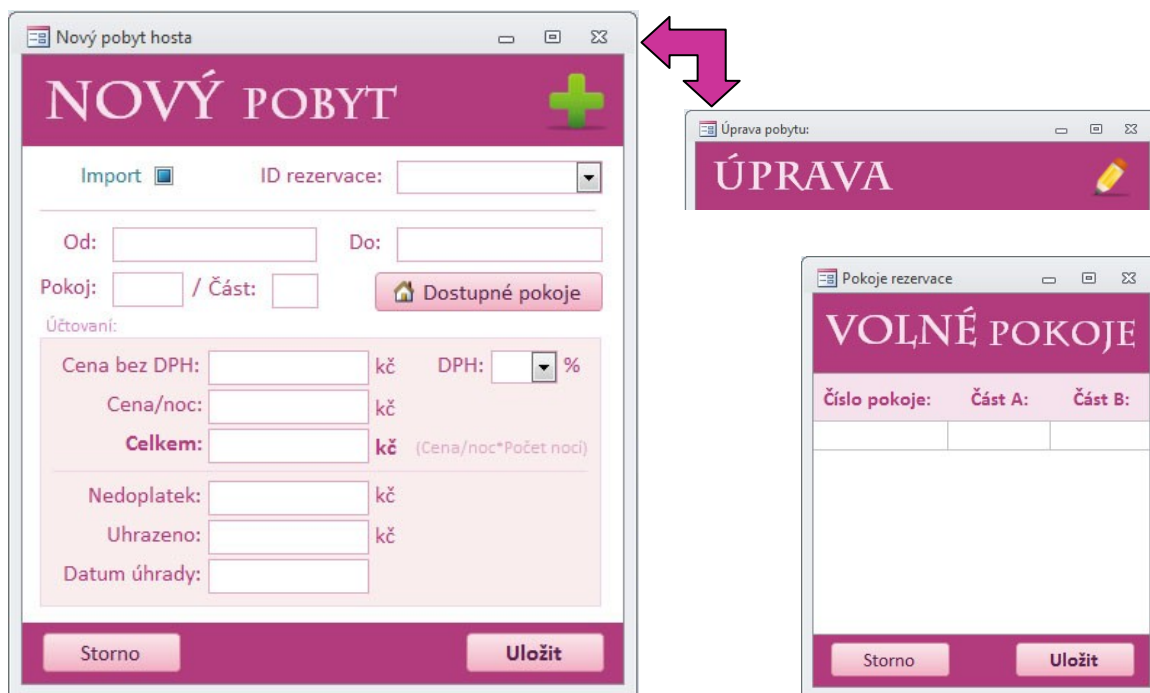
Obrázok 19 – Nový hosť, úprava hosťa

Každý z hosťov zaznamenaných v knihe ubytovaných má ďalej k sebe priradené svoje pobyty, teda ich kompletný prehľad, spolu s detailmi. Tieto pobyty môžeme opäť vytvárať, mazať, editovať alebo ukončiť. Pobyty sa dajú opäť filtrovať podľa toho, či sú uhradené alebo nie. Všetky tieto operácie vlastne tvoria spomínaný modul **Pobyty**, vid'. Obr.20

Pri vytváraní pobytu je podobne ako pri vytvorení nového hosťa možné, okrem ručného vypísania údajov, využiť aj ich import z niektorej rezervácie. Importujú sa dátumy začiatku a konca pobytu, podľa dátumov v tejto rezervácii. Počet nocí sa vyplní automaticky, na základe rozsahu zadáných dátumov. Po vytvorení má pobyt status *prebieha*, keďže toto vytvorenie sa uskutoční až po príchode hosťa a jeho ubytovaní. Jeden hosť môže mať viacero pobytov pod jednou rezerváciou, nemôžu však prebiehať súčasne. Fakt, že hosť môže mať viacero pobytov pod jednou rezerváciou môže spočiatku vypadať nelogicky, no táto možnosť v systéme existuje najmä kvôli rezerváciám, ktoré uskutočňujú firmy pre svojich zamestnancov. V takomto prípade sa často stáva, že firma napríklad rezervuje jednu izbu na mesiac a počas tohto mesiaca sa v nej po týždni striedajú dvaja robotníci. Každý má teda dva pobyty pod jednou rezerváciou. Z tohto dôvodu tvorí primárny kľúč tabuľky *tblPobyty* okrem čísla občianskeho preukazu a ID rezervácie aj dátum začiatku rezervácie.

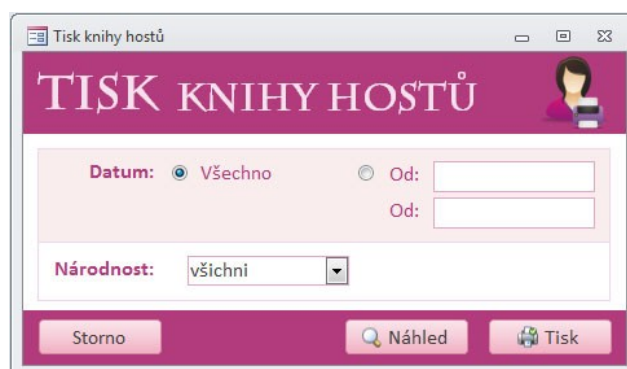
Pri ukončení pobytu ostáva tento pobyt uložený v tabuľke pobytov (*tblPobyty*) a jeho status sa mení na *ukončený*, naopak pri odstránení pobytu hosťa, je tento pobyt z tabuľky zmazaný. *CisloPokoje* v tabuľke *tblPobyty* nie je cudzím kľúčom z tabuľky *tblPokoje*, hoci ho reprezentuje, kvôli tomu, aby pri zmazaní záznamu z *tblPokoje* neboli ovplyvnené pobyty v ktorých sa tento záznam resp. pokoj vyskytoval. Zjednodušene : Chceme uchovávať históriu pobytov nezávisle na tom, či izba, ktorá bola pobytu priradená momentálne existuje alebo nie. Toto však platí len pre *ukončené* pobyty. Pri odstránení izby musíme overiť, či sa daná izba nevyskytuje u niektorého *aktuálneho* pobytu. V takomto prípade musí byť tento pobyt ukončený alebo mu musíme zmeniť izbu.

Obrázok 20 – Modul: *Pobyty hosta*



Obrázok 21 – Nový pobyt, úprava pobytu, voľné pokoje

Ako už bolo spomenuté kniha ubytovaných má byť dostupná aj v papierovej verzii, takže je pridaná možnosť tlače podľa pripravenej tlačovej zostavy. Používateľ si môže prostredníctvom jednoduchého formulára špecifikovať, či sa má vytlačiť celá kniha, alebo len záznamy v určitom rozpätí dátumov. Možno je tiež vytlačiť záznamy na základe národnosti host'a resp. toho, či je cudzinec, alebo nie. Používateľovi je pred tlačou umožnené zobrazíť aj náhľad dokumentu.



Obrázok 22 – Tlač knihy ubytovaných

Adresář

Tento modul slúži na uchovávanie a správu údajov o firmách a živnostníkoch. Záznamy o firmách môžeme vytvárať, mazať, upravovať, či prehľadávať. Sú zobrazené podobne ako pri ostatných formulároch, teda v jednej časti formuláru je ich zoznam a v druhej detaily, ktoré sa zobrazia po kliknutí naň. (Obr.23)

Pri vytváraní, editovaní, alebo zobrazení detailu záznamu máme možnosť skontrolovať, či je zadané identifikačné číslo firmy (IČ) pravé a prislúcha k danej firme. Kliknutím na tlačidlo sme odkázaný na webovú stránku živnostenského registra, kde sa po načítaní stránky automaticky vyplní pole s IČ firmy.

The screenshot shows the 'Adresář' (Address Book) module. On the left, there is a table listing companies with columns for Name, IČ (Identification Number), and City. The table contains several entries, including 'JaT' with IČ 12345678 and City Turzovka. On the right, the detailed view for 'JaT' is shown, including its address (Mladeže, Turzovka, Slovensko), IČ, DIČ, Účet (090909 / 0001), Email (jat@zoznam.sk), Tel. 1 (4353444), Mobil (333989333), and Fax (837687635). Below the table, there are buttons for 'Menu', 'Smazať', and 'Upraviť'. A blue arrow points from the 'Upraviť' button to the 'NOVÝ KONTAKT' form below.

NOVÝ KONTAKT

Název:

Adresa:

Ulice:

Město: PSČ:

Země:

IČ: DIČ:

Účet: / Kod. b:

Kontakt:

Email:

Tel. 1: Mobil:

Tel. 2: Fax:

Poznámka:

ÚPRAVA

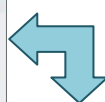
Obrázok 23 – Modul: Adresář, nový kontakt, upraviť kontakt

Faktury

V systéme je tiež dostupná správa faktúr. Firma vedie iba odberateľské faktúry. Môžeme ich opäť klasicky vytvárať, mazat', editovať, vyhľadávať a tlačiť. Pri tvorbe a editácii faktúr je možné využiť import údajov buď z *Adresáře* firiem, alebo *Knihy hostů*, podľa toho komu je faktúra vystavená. Podľa tabuľky *tblFaktury* môžu byť obaja odberatelia *null*, no pomocou validácie vo formulári zabezpečíme aby bol zadaný aspoň jeden z nich. Pri tlači je tiež možné zobrazit' náhľad.

The screenshot shows a web application window titled 'FAKTURY'. At the top, there's a header with the title and a 'Nová' button. Below it is a table with columns: Číslo faktury, Odběratel, Datum vystavení, Celkem k úhradě, Datum splatnosti, Forma úhrady, and Uhrazeno. The first row is labeled '(Nové)'. Below the table is a search bar with 'Záznam: 1 z 1' and a 'Vyhledávání' button. The bottom section contains a summary with labels: Číslo faktury, Odběratel, Vystavil, Cena bez DPH, DPH, Celkem k úhradě, and Záloha, each followed by a text input field and a 'Kč' or '%' unit. At the very bottom, there are buttons for 'Menu', 'Smazat', 'Upravit', 'Náhled', and 'Tisk'.

The screenshot shows a web application window titled 'NOVÁ FAKTURA'. It contains several input fields for creating a new invoice. The 'Odběratel' section has fields for 'IČ:' and 'OP:', with buttons for 'Adresář' and 'Kniha h.'. The 'Platební podmínky' section has fields for 'Forma úhrady:', 'DUZP:', 'Číslo účtu:', 'Datum vystavení:', 'Konstantní symbol:', and 'Datum splatnosti:'. The 'Fakturovaná položka' section has fields for 'Datum od:', 'Datum do:', 'Počet nocí:', and 'Text:'. At the bottom, there are fields for 'Cena bez DPH:', 'DPH:', 'Záloha:', and 'Celkem k úhradě:', each with a unit ('Kč' or '%'). There are also buttons for 'Storno', 'Náhled', 'Tisk', and 'Uložit'.



The screenshot shows a web application window titled 'ÚPRAVA'. It has a header with the title and a button with a pencil icon. The rest of the form is not visible in this snippet.

Obrázok 24 – Modul: *Faktury*, nová faktúra, upraviť faktúru

Pokladní doklady

Systém eviduje na obdobnom princípe ako u faktúr aj pokladničné doklady. Je ich teda možné vytvárať, mazat', editovať, vyhľadávať a tlačit'. Súčasťou vytvárania a editácie pokladničných dokladov je taktiež možnosť importovať údaje z *Adresáře* firiem, alebo *Knihy hostů*. Jediný rozdiel u pokladničných dokladov je ten, že si firma uchováva doklady príjmové aj výdajové. Zobrazenie jednotlivých pokladničných dokladov v module je zabezpečené pomocou prepínacích tlačidiel (*Príjmové/Výdajové*).

Číslo dokladu	Přijato od	Datum vystavení	Celkem	Cena bez DPH	DPH %	Částka DPH
(Nové)						

Číslo dokladu: _____ Datum vystavení: _____
Přijato od: _____ Cena bez DPH: _____ Kč
IČ: _____ DPH: _____ %
Účel platby: _____ Celkem: _____ Kč
Slovy: _____

Menu Smazat Upravit Náhled Tisk

NOVÝ PPD

Přijato od IČ: _____ OP: _____
Účel platby: _____
Datum: _____
Cena bez DPH: _____ Kč DPH: _____ %
Celkem: _____ Kč
Slovy: _____

Adresář Kniha h. Storno Náhled Tisk Uložit



Úprava PPD: ÚPRAVA

Obrázok 25 – Modul: *Pokladní doklady*, nový PPD, upraviť PPD

Poplatky

V rámci aplikácie nebolo implementované riešenie na spracúvanie poplatkov. V menu aplikácie sa nachádza len odkaz na externý súbor MS Excel, kde firma poplatky spracúva.

4 Zhodnotenie navrhovaného riešenia

Systém je navrhnutý na základe špecifických požiadaviek firmy K-Service, takže jeho využiteľnosť, bez ďalších úprav, u inej firmy je otázná. Na druhú stranu je takéto „riešenie šité na mieru“ veľmi silným resp. vhodným nástrojom pre menšie firmy, keďže tie by len ťažko využili všetku funkcionality existujúcich komerčných produktov, ktoré by tak zbytočne preplácali. Najväčšou výhodou pri použití vlastného riešenia je teda to, že nám umožňuje sústrediť sa na veci špecifické pre danú firmu.

Náš návrh zjednocuje a optimalizuje inak roztrieštené moduly systému, ktoré boli spravované rôznymi spôsobmi.

Ďalej prikladám zhodnotenie návrhu systému slovami spolumajiteľky firmy K-service, pani Renáty Kapusňákovvej. Scan prehlásenia je priložený v prílohe č. 7.

„S celkovým návrhom systému sme spokojní. Používateľské prostredie aplikácie je veľmi príjemné a jednoduché, úplne prevýšilo naše očakávania. Návrh funkčnosti spĺňa všetky naše požiadavky. Po doplnení samotnej funkčnosti plánujeme nasadiť systém do testovacej prevádzky, kde by boli odhalené prípadné nedostatky a neskôr po odladení všetkých chýb, by sme ho radi použili v reálnej prevádzke. Spolupráca so študentom prebiehala bez problémov a celkovo ju hodnotíme veľmi kladne.“ Renáta Kapusňáková, spolumajiteľka firmy K-Service

5 Záver

Cieľom tejto práce bola racionalizácia databázového systému firmy K-Service, ktorá vlastní ubytovacie zariadenie v Prahe a hotel vo Veľkých Popoviciach. Pri analýze súčasného stavu vedenia databázy vo firme sa zistilo, že firma používa nepraktický a neucelený, manuálne-elektronický spôsob správy dát. Mojou úlohou teda bolo navrhnúť úplne nový, ucelený databázový systém, na základe požadovanej funkcionality, v prostredí MS Access 2010.

Návrh systému v sebe zahŕňal vytvorenie designu používateľského rozhrania (UI), ktoré bolo koncipované tak, aby bolo jednoduché, prehľadné a zároveň príjemné na obsluhu. Firma bola s finálnym UI veľmi spokojná. Design celého systému bol samozrejme úzko spätý s jeho funkčnosťou. Tá bola detailne analyzovaná a vyjadrená pomocou rôznych diagramov ako napr. use case diagramy. Pri jej návrhu boli zohľadnené a splnené všetky požiadavky firmy. Funkčnosť je v systéme implementovaná len z časti, pre ukážku. Dôraz bol kladený hlavne na jej podrobnú analýzu, podľa ktorej bolo potom možné navrhnúť UI a databázu. Samotná logika aplikácie je, okrem už spomenutého UI, podporená aj vytvorenou databázou. Pri analýze funkčnosti a databázy vzniklo viacero problémov, ktoré sa však podarilo vyriešiť.

Ako vylepšenie systému vidím hlavne doimplementovanie chýbajúcej funkcionality, ďalej pridanie databázových tabuliek, ktoré by slúžili na ukladanie histórie záznamov, keďže teraz ostávajú záznamy po „ukončení“ v rovnakej tabuľke v akej boli, mení sa len ich status. Okrem toho by bolo tiež vhodné spracovať lepšie zobrazenie celkového prehľadu hotela, napr. s využitím kalendára.

Pri tvorbe tejto práce som si zdokonalila vedomosti týkajúce sa práce s programom MS Access a taktiež vedomosti ohľadom databáz všeobecne. Najväčším prínosom však pre mňa bolo hlavne prežitie si časti cyklu vývoja softwaru, od analýzy požiadaviek až po čiastočnú implementáciu jednotlivých modulov spolu s komunikáciou so zákazníkom.

Na záver by som chcela povedať, že aplikáciu by som rada v spolupráci s firmou K-service kompletne dokončila a uviedla do reálnej prevádzky.

Zoznam použitej literatúry

Knihy:

- [1] VIESCAS, John a Jeff CONRAD. *Microsoft Office Access 2007 Inside Out*. Redmond (Washington): Microsoft Press, 2007. ISBN 0-7356-2325-2.
- [2] KRUCZEK, Aleš. *Microsoft Office Access 2010: Podrobná uživatelská příručka*. Brno: Computer Press, 2010. ISBN 978-80-251-3289-0.

Internetové zdroje:

- [3] Policie České republiky – Služba cizinecké policie. *policie.cz*. [Online]. (c) 2010, [cit. 2012-4-24]. Dostupné z: <http://www.policie.cz/clanek/hlaseni-pobytu-cizincu.aspx>.
- [4] Základy databáz. *office.microsoft.com*. [Online]. (c) 2012, [cit. 2012-4-24]. Dostupné z: <http://office.microsoft.com/sk-sk/access-help/zaklady-databaz-HA010064450.aspx>.
- [5] Funkcie a výhody databázového softvéru Access 2010. *office.microsoft.com*. [Online]. (c) 2012, [cit. 2012-4-24]. Dostupné z: <http://office.microsoft.com/sk-sk/access/funkcie-a-vyhody-databazoveho-softveru-access-2010-HA101809011.aspx>.
- [5] SKŘIVAN, Jaromír. Databáze nejsou jen MySQL. *interval.cz*. [Online]. 2002, [cit. 2012-4-24]. Dostupné z: <http://interval.cz/clanky/databaze-nejsou-jen-mysql/>.
- [6] SKŘIVAN, Jaromír. Databáze a jazyk SQL. *interval.cz*. [Online]. 2000, [cit. 2012-4-1]. Dostupné z: <http://interval.cz/clanky/databaze-a-jazyk-sql/>.
- [7] KULHAN, Jakub. Normalizace relačních databází. *programujte.com*. [Online]. 2008, [cit. 2012-4-5]. Dostupné z: <http://programujte.com/clanek/2008071900-normalizace-relacnich-databazi/>.
- [8] Teorie relačních databází: Normalizace. *manualy.net*. [Online]. 2007, [cit. 2012-4-5]. Dostupné z: <http://www.manualy.net/article.php?articleID=13>.
- [9] Teorie relačních databází: Integritní omezení. *manualy.net*. [Online]. 2006, [cit. 2012-4-5]. Dostupné z: <http://www.manualy.net/article.php?articleID=15>.

[10] Teorie relačních databází: Relační model dat. *manualy.net*. [Online]. 2006, [cit. 2012-4-5]. Dostupné z: <http://www.manualy.net/article.php?articleID=9>.

[11] Databáza – základné pojmy. *spsepn.edu.sk*. [Online]. [cit. 2012-4-24]. Dostupné z: <http://www.spsepn.edu.sk/bc/index.php?stranka=programovanie&kap=www&kat=datab>.

[12] What is Microsoft Access. *simply-access.com*. [Online]. (c) 2012 , [cit. 2012-4-22]. Dostupné z: <http://www.simply-access.com/What-Is-Microsoft-Access.html>.

[13] Visual Basic for Applications. *harcek.com*. [Online]. 2011, [cit. 2012-4-22]. Dostupné z: <http://www.harcek.com/content/visual-basic-applications>.

[14] Hotelový software: Horec. *hotelovesystemy.cz*. [Online]. (c) 2002 - 2005, [cit. 2012-4-24]. Dostupné z: <http://www.hotelovesystemy.cz/systemy-pro-hotely/hotelovy-software.php?id=117&firma=Horec>.

[15] Hotelový software: Agnis. *hotelovesystemy.cz*. [Online]. (c) 2002 - 2005, [cit. 2012-4-24]. Dostupné z: <http://www.hotelovesystemy.cz/systemy-pro-hotely/hotelovy-software.php?id=78&firma=Agnis>.

Prednášky:

[16] NOVÁK, Vítězslav. Návrh databázové atabázové aplikace. Prednášky k predmetu. Dostupné z: <https://portalekf.wps.vsb.cz/wps/portal>.

Zoznam skratiek

DML	– Data Manipulation Language
DDL	– Data Definition Language
DBS	– Database System
DB	– Database
DBMS	– Database Management System
ER	– Entitno-Relačný
ERD	– Entitno-Relačný Diagram
GUI	– Graphical User Interface
IO	– Integritné Obmedzenie
MS	– Microsoft
RDBMS	– Relational Database Management System
RMD	– Relačný Model Dát
SDL	– Storage Definition Language
SRBD	– Systém Riadenia Báže Dát
SQL	– Structured Query Language
UI	– User Interface
VBA	– Visual Basic For Applications
VDL	– View Definition Language

Zoznam obrázkov

Obrázok 1 – Ukážka interface MS Access 2010.....	19
Obrázok 2 – Ukážka tvorby ikon v programe CorelDraw	25
Obrázok 3 – Ukážka ikon.....	26
Obrázok 4 – Kombinácie farieb	26
Obrázok 5 – Use case diagram (úroveň 1)	27
Obrázok 6 – Diagram vzťahov entít.....	28
Obrázok 7 – Prihlasovací formulár	30
Obrázok 8 – Registrácia používateľa	30
Obrázok 9 – Menu aplikácie	31
Obrázok 10 – Diagram aktivít: Správa izieb	32
Obrázok 11 – Modul: <i>Pokoje</i>	33
Obrázok 12 – Úprava izby	33
Obrázok 13 – Nová izba.....	33
Obrázok 14 – Modul: <i>Rezervace</i>	35
Obrázok 15 – Nová rezervácia, úprava rezervácie a pokoje rezervácie.....	36
Obrázok 16 – Stavový diagram: Rezervácie	36
Obrázok 17 – Modul: <i>Přehled hotelu</i>	37
Obrázok 18 – Modul: <i>Kniha hostů</i>	38
Obrázok 19 – Nový hosť, úprava host'a	39
Obrázok 20 – Modul: <i>Pobyty hosta</i>	40
Obrázok 21 – Nový pobyt, úprava pobytu, voľné pokoje.....	41
Obrázok 22 – Tlač knihy ubytovaných	41
Obrázok 23 – Modul: <i>Adresář</i> , nový kontakt, upraviť kontakt.....	42
Obrázok 24 – Modul: <i>Faktury</i> , nová faktúra, upraviť faktúru	43
Obrázok 25 – Modul: <i>Pokladní doklady</i> , nový PPD, upraviť PPD.....	44

Zoznam tabuliek

Tabuľka 1 – Dátový slovník tabuľky <i>tblKnihaHostu</i>	29
---	----

Prehlásenie o využitíu výsledkov bakalárskej práce

Prohlašuji, že

- jsem byl(a) seznámen(a) s tím, že na mou diplomovou (bakalářskou) práci se plně vztahuje zákon č. 121/2000 Sb. – autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo;
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen VŠB-TUO) má právo nevýdělečně, ke své vnitřní potřebě, diplomovou (bakalářskou) práci užít (§ 35 odst.3);
- souhlasím s tím, že diplomová (bakalářská) práce bude v elektronické podobě archivována v Ústřední knihovně VŠB-TUO a jeden výtisk bude uložen u vedoucího diplomové (bakalářské) práce. Souhlasím s tím, že bibliografické údaje o diplomové (bakalářské) práci budou zveřejněny v informačním systému VŠB-TUO;
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 autorského zákona;
- bylo sjednáno, že užít své dílo, diplomovou (bakalářskou) práci, nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).

V Ostravě dne 11.5.2012

.....
Dominika Gajdičiarová

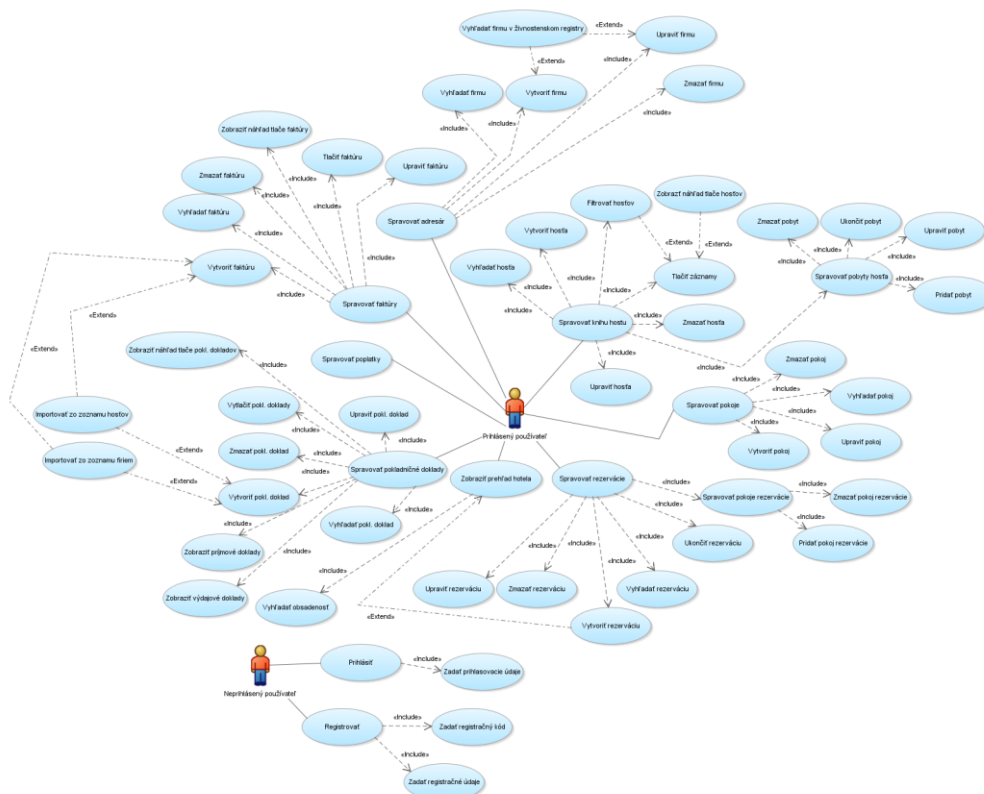
Zoznam príloh

- Príloha č. 1 – Use case diagram 2
- Príloha č. 2 – Use case diagram 3
- Príloha č. 4 – Stavový diagram: Pobyty
- Príloha č. 4 – Faktura
- Príloha č. 5 – Příjmový pokladní doklad
- Príloha č. 6 – Výdajový pokladní doklad
- Príloha č. 7 – Vyjadrenie spoločnosti

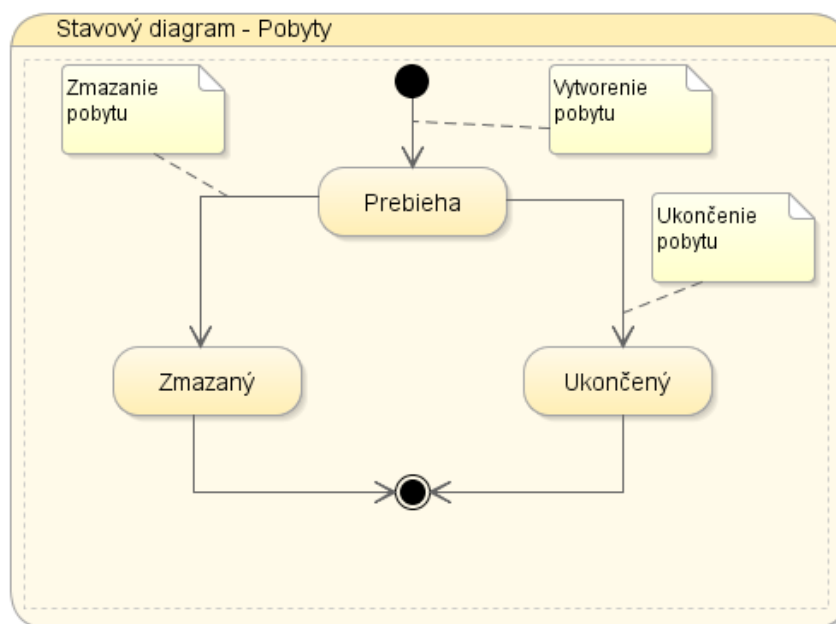
Príloha č. 1



Príloha č. 2




Príloha č. 3



FAKTURA - DAŇOVÝ DOKLAD

Číslo faktury:

Dodávateľ: K-Service  Nad Opatovcem 2140/2 149 00 Praha 4 IČ: 76240070 DIČ: CZ5460142424 Telefón: 736 425 101 E-mail: kapusneko@volny.cz		Odběratel:													
Platební podmínky: Forma úhrady: Datum uskutečnění plnění: Číslo účtu: Datum vystavení: Variabilní symbol: Konstantní symbol: Datum splatnosti															
Fakturované položky: Fakturujeme Vám: za ubytování od do počet nocí:		Cena bez DPH Sazba DPH Cena s DPH													
Zaplacená záloha Kč: Celkem k úhrade Kč:															
<table border="1"> <thead> <tr> <th>Sazba</th> <th>Zaokrouhlení</th> <th>Základ daně</th> <th>Výše daně</th> </tr> </thead> <tbody> <tr> <td>0 %</td> <td>0,00</td> <td></td> <td></td> </tr> <tr> <td>14 %</td> <td>0,00</td> <td></td> <td></td> </tr> </tbody> </table> <p>Důležité upozornění: upozornit, že případě nedodržení data splatnosti Vám budeme účtovat úrok z prodlení ve výši 0,05% ze každý den z prodlení.</p>				Sazba	Zaokrouhlení	Základ daně	Výše daně	0 %	0,00			14 %	0,00		
Sazba	Zaokrouhlení	Základ daně	Výše daně												
0 %	0,00														
14 %	0,00														
Vystavil: Razítko a podpis:															

Príloha č. 5

PŘÍJMOVÝ pokladní doklad		č:	ze dne
Irena KAPUSŇÁKOVÁ Nad Opatovem 2140/2 149 00 Praha 4 IČ: 76240070 DIČ: CZ5460142424		Cena bez DPH	
		+ % DPH	
		Celkem Kč	
Slovy			
Přijato od:		IČ:	
Účel platby:			
Převzal:		Zaúčtoval:	

Príloha č. 6

VÝDAJOVÝ pokladní doklad		č:	ze dne
Irena KAPUSŇÁKOVÁ Nad Opatovem 2140/2 149 00 Praha 4 IČ: 76240070 DIČ: CZ5460142424		Cena bez DPH	
		+ % DPH	
		Celkem Kč	
Slovy			
Vyplaceno komu:		IČ:	
Účel platby:			
Převzal:		Zaúčtoval:	

S celkovým návrhom systému sme spokojní. Používateľské prostredie aplikácie je veľmi príjemné a jednoduché, úplne prevýšilo naše očakávania. Návrh funkčnosti spĺňa všetky naše požiadavky. Po doplnení samotnej funkčnosti plánujeme nasadiť systém do testovacej prevádzky, kde by boli odhalené prípadné nedostatky a neskôr po odladení všetkých chýb, by sme ho radi použili v reálnej prevádzke. Spolupráca so študentom prebiehala bez problémov a celkovo ju hodnotíme veľmi kladne.

HOTEL SLAVOJ
Ringhofferova 336
251 69 VELKÉ POPOVICE
IČO: 76240070, DIČ: CZ5460142424
tel./fax: 323 665 226 / 732 238 189